NetBackup™ Deduplication Guide

UNIX, Windows, and Linux

Release 11.2.0.1

COHESITY

NetBackup™ Deduplication Guide

Last updated: 2026-09-09

Legal Notice

Copyright © 2026 Cohesity, Inc. All rights reserved.

© 2026 Cohesity, Inc. All Rights Reserved. Cohesity, the Cohesity Logo and other Cohesity Marks are trademarks of Cohesity, Inc. in the US and/or internationally. The information supplied herein is the confidential and proprietary information of Cohesity and may only be used (a) by the intended recipients and (b) in conjunction with validly licensed Cohesity software and services. Find the terms of Cohesity licenses at www.cohesity.com/agreements.

THE DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID. COHESITY SHALL NOT BE LIABLE FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS DOCUMENTATION. THE INFORMATION CONTAINED IN THIS DOCUMENTATION IS SUBJECT TO CHANGE WITHOUT NOTICE.

Cohesity Support

Reach Cohesity Support

There are several ways to create a Cohesity support case.

- Go to Cohesity Support, to search in our knowledge base; or contact us by phone - United States and Canada: 1-855-9CO-HESI (926-4374), option 2.
- Log in to the Cohesity Support Portal to create a new case.
- Click the (**?**) icon on the Cohesity UI and select Support Portal.

Support/Service Assistance

First, contact the Service Provider that you have contracted for service and support. If you work directly with Cohesity and have a product warranty/entitlement, repair pricing, or technical support-related question, see your options below:

- To find solutions to your product issues or for suggestions or best practices, visit the Cohesity Knowledge Base.
- Log in to the Cohesity Support Portal to create a new case.
- To monitor your open cases, log in to the portal and click the **Cases** tab on the home page. This page should have all the case statuses and updates. You can also view individual case status.

Cohesity Software Running on Partner Hardware

For Cohesity software running on qualified third-party hardware, the following support workflow applies:

1. The customer may contact Cohesity Support first if the issue cannot be determined as a hardware issue.

   Note: Cohesity cannot process hardware replacement requests for partner hardware.

2. Cohesity Support triages the issue. If it is a software issue, Cohesity Support continues to work on it.
3. If it is a hardware/firmware issue or is suspected to be a hardware/firmware issue, Cohesity provides information about the issue to the customer and requests that the customer open a support ticket with the appropriate partner.
4. If needed, Cohesity Support can join a three-way call with the partner and the customer.
5. The customer informs Cohesity Support on the progress of the partner's case.

Contents

Chapter 1 Introducing the NetBackup media server deduplication option 22

About the NetBackup deduplication options 22

Chapter 2 Planning your deployment 24

Planning your MSDP deployment 25
NetBackup naming conventions 26
About MSDP deduplication nodes 27
About the NetBackup deduplication destination 27
About MSDP capacity support and hardware requirements 28
About MSDP storage and connectivity requirements 30
Fibre Channel and iSCSI comparison for MSDP 32
About NetBackup media server deduplication 34
About MSDP storage servers 35
About MSDP load balancing servers 36
About MSDP server requirements 37
About MSDP unsupported configurations 38
About NetBackup Client Direct deduplication 38
About MSDP client deduplication requirements and limitations 40
About MSDP remote office client deduplication 40
About MSDP remote client data security 41
About remote client backup scheduling 41
About the NetBackup Deduplication Engine credentials 41
About the network interface for MSDP 42
About MSDP port usage 43
About MSDP optimized synthetic backups 44
About MSDP and SAN Client 45
About MSDP optimized duplication and replication 45
About MSDP stream handlers 46
Oracle stream handler 46
Microsoft SQL Server stream handler 48
MSDP deployment best practices 50
Use fully qualified domain names 51
About scaling MSDP 51

Send initial full backups to the storage server 52
Increase the number of MSDP jobs gradually 52
Introduce MSDP load balancing servers gradually 52
Implement MSDP client deduplication gradually 53
Use MSDP compression and encryption 53
About the optimal number of backup streams for MSDP 53
About storage unit groups for MSDP 54
About protecting the MSDP data 54
Save the MSDP storage server configuration 55
Plan for disk write caching 55

Chapter 3 Provisioning the storage 57

About provisioning the storage for MSDP 57
Do not modify MSDP storage directories and files 59
About volume management for NetBackup MSDP 59

Chapter 4 Configuring deduplication 61

Configuring media server deduplication in NetBackup 63
Configuring MSDP client-side deduplication 65
About the MSDP Deduplication Multi-Threaded Agent 66
Configuring the Deduplication Multi-Threaded Agent behavior 67
MSDP mtstrm.conf file parameters 68
Configuring deduplication plug-in interaction with the Multi-Threaded Agent 72
About MSDP fingerprinting 73
About the MSDP fingerprint cache 73
Configuring the MSDP fingerprint cache behavior 74
MSDP fingerprint cache behavior options 74
About the last backups of clients 77
About seeding the MSDP fingerprint cache for remote client deduplication 81
Configuring MSDP fingerprint cache seeding on the client 83
Configuring MSDP fingerprint cache seeding on the storage server 84
NetBackup seedutil options 85
About sampling and predictive cache 86
Rebuilding the sampling cache 88
Enabling 400 TB support for MSDP 89
Creating the data directories for 400 TB MSDP support 89
Adding volumes to a 400 TB Media Server Deduplication Pool 90

Configuring a storage server for a Media Server Deduplication Pool 93
MSDP storage path properties 95
MSDP network interface properties 97
About disk pools for NetBackup deduplication 97
Configuring a disk pool for deduplication 98
Media Server Deduplication Pool properties 100
Configuring a Media Server Deduplication Pool storage unit 102
Media Server Deduplication Pool storage unit properties 102
MSDP storage unit recommendations 104
Configuring client attributes for MSDP client-side deduplication 106
Disabling MSDP client-side deduplication for a client 107
Disable client-side deduplication for all clients in a policy 107
About MSDP compression 108
About MSDP encryption 109
Configuring encryption for MSDP local storage volume 109
Configuring encryption for MSDP cloud storage volumes 110
Configuring MSDP encryption on different platforms 111
Modes of rolling data conversion 111
MSDP encryption behavior and compatibilities 113
About MSDP Encryption using NetBackup Key Management Server service 114
Enabling KMS encryption for a local LSU 116
Upgrading KMS for MSDP 117
About MSDP Encryption using external KMS server 119
Configuring optimized synthetic backups for MSDP 120
About a separate network path for MSDP duplication and replication 120
Configuring a separate network path for MSDP duplication and replication 121
About MSDP optimized duplication within the same domain 122
About the media servers for MSDP optimized duplication within the same domain 124
About MSDP push duplication within the same domain 125
About MSDP pull duplication within the same domain 128
Configuring MSDP optimized duplication within the same NetBackup domain 129
Configuring NetBackup optimized duplication or replication behavior 133
Setting NetBackup configuration options by using the command line 135
About MSDP replication to a different domain 136
Configuring MSDP replication to a different NetBackup domain 137

About NetBackup Auto Image Replication 139
About trusted primary servers for Auto Image Replication 146
About the certificate to use to add a trusted primary server 150
Add a trusted primary server 151
Remove a trusted primary server 152
Enable inter-node authentication for a NetBackup clustered primary server 153
Configuring NetBackup CA and NetBackup host ID-based certificate for secure communication between the source and the target MSDP storage servers 154
Configuring external CA for secure communication between the source MSDP storage server and the target MSDP storage server 155
Configuring a target for MSDP replication to a remote domain 156
About configuring MSDP optimized duplication and replication bandwidth 160
About performance tuning of optimized duplication and replication for large images 162
About performance tuning of optimized duplication and replication for MSDP cloud 162
About storage lifecycle policies 162
About the storage lifecycle policies required for Auto Image Replication 163
Creating a storage lifecycle policy 165
Storage lifecycle policy settings 166
About MSDP backup policy configuration 169
Resilient network properties 170
Resilient connection resource usage 172
Specify resilient connections for clients 172
Adding an MSDP load balancing server 174
About variable-length deduplication on NetBackup clients 174
Managing the variable-length deduplication using the cacontrol command-line utility 176
About the MSDP pd.conf configuration file 178
Editing the MSDP pd.conf file 179
MSDP pd.conf file parameters 179
About the MSDP contentrouter.cfg file 194
About saving the MSDP storage server configuration 195
Saving the MSDP storage server configuration 196
Editing an MSDP storage server configuration file 197
Setting the MSDP storage server configuration 198
About the MSDP host configuration file 199

Deleting an MSDP host configuration file 200
Resetting the MSDP registry 200
About protecting the MSDP catalog 201
About the MSDP shadow catalog 201
About the MSDP catalog backup policy 202
Changing the MSDP shadow catalog path 204
Changing the MSDP shadow catalog schedule 205
Changing the number of MSDP catalog shadow copies 206
Configuring an MSDP catalog backup 207
MSDP drcontrol options 208
Updating an MSDP catalog backup policy 211
About MSDP FIPS compliance 212
Configuring the NetBackup client-side deduplication to support multiple interfaces of MSDP 215
About MSDP multi-domain support 215
About MSDP application user support 220
About MSDP mutli-domain VLAN Support 221
About NetBackup WORM storage support for immutable and indelible data 224
About the NetBackup command line options to configure immutable and indelible data 225
Running MSDP services with the non-root user 227
Changing the service user after installation or upgrade 227
Running MSDP commands with the non-root user 229
Configuring a non-root user to run MSDP commands through msdpcmdrun 230
About PDPAS logging 231
Sample msdpcmdrun commands 232
Considerations for file access 232
Running the MSDP commands on NetBackup appliances 232

Chapter 5 MSDP volume group (MVG) 233

About the MSDP volume group 234
The MSDP volume group components 235
About MVG resilience 236
Configuring the MSDP volume group 236
MSDP volume group requirements 237
Configuring an MVG server using the web UI 239
Creating an MVG volume using the web UI 239
Configuring an MVG server using the command-line 240
Creating an MVG volume using the command-line 242
Updating an MVG volume using the command-line 243

Configuring the targeted AIR with an MVG volume 244
Updating an MVG volume using the web UI 244
Listing the MVG volumes 245
Deleting an MVG volume 245
Deleting a disk volume from an MVG volume configuration 246
Configuring the MSDP server to be used by an MVG server having different credentials 248
Migrate a backup policy from a regular MSDP disk volume to the MVG volume 250
Migrate a backup policy from an MVG volume to a regular MSDP disk volume 250
Assigning a client policy combination to another MSDP server 251
Removing an MVG server configuration 251
MSDP volume group disaster recovery 252
The MSDP server maintenance 254
Limitations of the MSDP volume group 255
About the node failure management 256
MSDP volume group best practices 258
MSDP commands for MVG maintenance 259
Troubleshooting the MVG errors 262
Examples of valid WORM MVG volumes 264
About the client and policy assignments with disk volumes in MVG volume 265
The modes of a disk volume in MVG volume 267
Recommendation for managing the data distribution 268
MVG load balancing levels 269
Specifications and default values of the MVG tuning parameters 270
CLI for tuning MVG parameters 272

Chapter 6 MSDP server evacuation 273

About MSDP server evacuation 273
Prerequisites for MSDP server evacuation 273
Identifying images not created through the MVG volume on the MSDP server 275
Managing the MSDP server evacuation 276
Starting the MSDP server evacuation 276
Configuring the maximum concurrent number of jobs in the storage unit 277
Pausing or resuming the MSDP server evacuation 277
Changing the destination storage unit for MSDP server evacuation 278

Monitoring the MSDP server evaluation status 278
Handling the failed images gracefully 280
Cleaning up the images that are migrated from the MSDP server 281
Removing the disk volume of the MSDP server from the MVG volume configuration 282
Deleting the MSDP server configuration and decommissioning the MSDP/media server 282
Managing the MSDP server evacuation using the command-line 283
Starting the MSDP server evacuation using the command-line 284
Pausing the MSDP server evacuation using the command-line 285
Resuming the MSDP server evacuation using the command-line 286
Changing the destination storage unit using the command-line 287
Monitoring the status of MSDP server evacuation using the command-line 288
Cleaning up the images that are migrated from the MSDP server using command-line 289
Handling the failed images gracefully using the command-line 290
MSDP server evacuation recommendations 291
Migrating the cloud volume of an MVG volume 291

Chapter 7 MSDP cloud support 293

About MSDP cloud support 294
Operating system requirement for configuration 295
Limitations 295
Create a Media Server Deduplication Pool storage server 296
Managing credentials for MSDP-C 298
Creating a cloud storage unit 299
Updating cloud credentials for a cloud LSU 303
Updating encryption configurations for a cloud LSU 304
Deleting a cloud LSU 305
Backup data to cloud by using cloud LSU 307
Duplicate data cloud by using cloud LSU 307
Configuring AIR to use cloud LSU 308
About backward compatibility support 311
About the configuration items in cloud.json, contentrouter.cfg, and spa.cfg 312

Cloud space reclamation 318
Configuring the container aging 318
Configuring the cloud compaction 319
About the tool updates for cloud support 320
About the disaster recovery for cloud LSU 322
Common disaster recovery steps 331
Consolidating the cloud LSU 337
Disaster recovery for cloud LSU in Flex Scale 340
Additional steps for FortKnox Azure disaster recovery 341
About Image Sharing using MSDP cloud 342
Things to consider before you use image sharing to convert VM image to VHD in Azure 352
Converting the VM image to VHD in Azure 354
About restore from a backup in Microsoft Azure Archive 362
About FortKnox for NetBackup 362
Configuring FortKnox for NetBackup Azure and Azure Government 363
Configuring FortKnox for NetBackup Azure and Azure Government using the CLI 365
Configuring FortKnox for NetBackup Amazon and Amazon Government 374
Configuring FortKnox for NetBackup Amazon and Amazon Government using the CLI 375
Configuring FortKnox Google for NetBackup 381
Configuring FortKnox Google for NetBackup using the CLI 383
Migrating from standard authentication to token-based authentication for Recovery Vault 387
About MSDP cloud immutable (WORM) storage support 389
Creating a cloud immutable storage unit using the web UI 390
Updating a cloud immutable volume 391
About immutable object support for AWS S3 392
About immutable object support for AWS S3 compatible platforms 396
About immutable storage support for Azure blob storage 397
About object-level immutable storage support for Google Cloud Storage 398
About using the cloud immutable storage in a cluster environment 400
Troubleshooting the errors when disk volume creation using web UI fails 401
Deleting the immutable image with the enterprise mode 401
Deleting the S3 object permanently 402
About MSDP cloud admin tool 402

About AWS IAM Role Anywhere support 402
Prerequisites for AWS IAM Role Anywhere configuration 403
Configure IAM Role Anywhere in AWS 403
About Azure service principal support 408
Prerequisites for Azure service principal configuration 408
Configure Azure service principal 409
Configure a disk pool using Azure service principal 409
About instant access for object storage 410
Upgrading to NetBackup 10.3 and cluster environment 412
About the cloud direct 412
Configuring the cloud direct for backup 412
Configuring the cloud direct for restore 413
About MSDP lazy delete 414
About verification of the images stored in the archive tier 414

Chapter 8 S3 Interface for MSDP 415

About S3 interface for MSDP 415
Prerequisites for MSDP build-your-own (BYO) server 416
Configuring S3 interface for MSDP on MSDP build-your-own (BYO) server 417
Changing the certificate in S3 server 419
Changing the ETAG type of the S3 objects 419
Identity and Access Management (IAM) for S3 interface for MSDP 420
Signing IAM and S3 API requests 420
IAM workflow 420
IAM APIs for S3 interface for MSDP 423
IAM policy document syntax 446
S3 Object Lock In Flex WORM 449
S3 APIs for S3 interface for MSDP 450
S3 APIs on Buckets 451
S3 APIs on Objects 477
The naming rules for buckets and objects 502
Creating a protection policy for the MSDP object store 503
About MSDP S3 object import 504
How MSDP S3 object import works 505
Prerequisites for MSDP S3 object import 506
Configuring MSDP S3 object import 506
cacontrol --s3import command options 507
Limitations of the cacontrol --s3import command 508
Viewing MSDP S3 import jobs 508
Limitations of MSDP S3 object import 509

Troubleshooting MSDP S3 object import 509
Recovering the MSDP object store data from the backup images 510
Instant access for MSDP object store 510
Disaster recovery in S3 interface for MSDP 513
Recovering the MSDP S3 IAM configurations from cloud LSU 513
Limitations in S3 interface for MSDP 514
Logging and troubleshooting 514
Best practices 515

Chapter 9 Monitoring deduplication activity 517

Monitoring the MSDP deduplication and compression rates 517
Viewing MSDP job details 518
MSDP job details 519
About MSDP storage capacity and usage reporting 521
About MSDP container files 523
Viewing storage usage within MSDP container files 523
About monitoring MSDP processes 525
Reporting on Auto Image Replication jobs 525
Checking the image encryption status 526
MSDP events in NetBackup web UI 529

Chapter 10 Managing deduplication 530

Managing MSDP servers 531
Viewing MSDP storage servers 531
Determining the MSDP storage server state 531
Viewing MSDP storage server attributes 532
Setting MSDP storage server attributes 533
Changing MSDP storage server properties 534
Clearing MSDP storage server attributes 534
About changing the MSDP storage server name or storage path 535
Changing the MSDP storage server name or storage path 536
Removing an MSDP load balancing server 537
Deleting an MSDP storage server 538
Deleting the MSDP storage server configuration 539
Managing NetBackup Deduplication Engine credentials 539
Determining which media servers have deduplication credentials 540
Adding NetBackup Deduplication Engine credentials 540
Changing NetBackup Deduplication Engine credentials 541
Deleting credentials from a load balancing server 541

Managing Media Server Deduplication Pools 541
Viewing Media Server Deduplication Pools 542
Determining the Media Server Deduplication Pool state 542
Viewing Media Server Deduplication Pool attributes 542
Setting a Media Server Deduplication Pool attribute 543
Changing a Media Server Deduplication Pool properties 544
Clearing a Media Server Deduplication Pool attribute 548
Determining the MSDP disk volume state 549
Changing the MSDP disk volume state 550
Deleting a Media Server Deduplication Pool 550
Analyzing the disc space consumption of the backup images 551
Deleting backup images 552
About MSDP queue processing 552
Processing the MSDP transaction queue manually 553
About MSDP data integrity checking 554
Configuring MSDP data integrity checking behavior 555
MSDP data integrity checking configuration parameters 557
About the CRC notifications for local LSU 559
About managing MSDP storage read performance 560
About MSDP storage rebasing 561
MSDP server-side rebasing parameters 562
About the MSDP data removal process 563
Resizing the MSDP storage partition 564
How MSDP restores work 565
Configuring MSDP restores directly to a client 565
About restoring files at a remote site 566
About restoring from a backup at a target primary domain 566
Specifying the restore server 567
Enabling extra OS STIG hardening on WORM storage server instance 568
Using multiple MSDP nodes for multistream backups on MSDP cluster 569
Enabling the media server and MSDP engine affinity in the MSDP cluster 570

Chapter 11 Recovering MSDP 571

Recovering the NetBackup MSDP catalog 571
MSDP catalog recovery prerequisites 572
Synchronizing the date and time of the NetBackup primary server and the MSDP storage server 573
Determining the catalog backup for recovery 573
Recovering the MSDP catalog automatically 574

Recovering the MSDP Catalog from a backup 575
Recovering the MSDP catalog manually 575
Verifying the catalog backup files 576
Initializing MSDP for catalog recovery 578
Restoring an MSDP catalog backup to the catalog shadow directory 578
Recovering the MSDP Catalog from a shadow copy 579
Synchronize the MSDP catalog with the NetBackup catalog 579
Setting the NetBackup Deduplication Engine to refuse jobs 580
Checking MSDP data integrity manually 581
Process the MSDP transaction queue manually 582
Setting the NetBackup Deduplication Engine to accept jobs 582
Recovering from an MSDP storage server disk failure 583
Recovering from an MSDP storage server failure 584
Recovering the MSDP storage server after NetBackup catalog recovery 587

Chapter 12 Replacing MSDP hosts 588

Replacing the MSDP storage server host computer 588

Chapter 13 Uninstalling MSDP 591

About uninstalling MSDP 591
Deactivating MSDP 591

Chapter 14 Deduplication architecture 593

MSDP server components 593
Media server deduplication backup process 596
MSDP client components 597
MSDP client–side deduplication backup process 598

Chapter 15 Configuring and managing universal shares 601

Introduction to universal shares 601
Overview of universal shares 601
Key benefits 602
How to work with universal shares 603
Configuring and using an MSDP build-your-own (BYO) server for universal shares 603
Prerequisites to configure universal shares 604
Build-your-own (BYO) server prerequisites and hardware requirements to configure universal shares 604
Configuring universal share user authentication 607

Managing universal shares 617
Create a universal share 617
View or edit a universal share 626
Delete a universal share 627
Mounting a universal share 628
Creating a protection point for a universal share 630
Restoring data using universal shares 632
Restore methods of universal shares 633
Disaster recovery for a universal share 634
Disaster recovery for universal shares on migrated cloud LSU 635
Restore universal share data using Instant Access or single file recovery 637
Advanced features of universal shares 638
Direct universal share data to object store 639
Universal share accelerator for data deduplication 644
Ingest data to a universal share with the ingest mode 654
Take universal share snapshot over NFS or SMB 655
Trigger universal share backup over NFS or SMB 656
Universal share with disabled MSDP data volumes 657
Universal Share WORM capability 657
Universal share scale out 657
Managing universal share services 659
Scan data volumes with the vpfs_stats utility 659
Changing the number of vpfsd instances 661
Configuring the variable-length deduplication (VLD) algorithm for universal shares 663
Using the marker file interface for universal share operations 665
Troubleshooting issues related to universal shares 669
Troubleshooting universal share configuration issues 670
Logging and reporting for universal share VPFS instance 672
Vpfsd logs for file system operations in universal shares 672
Notify events to primary server with the vpfsd service 673
Data integrity check and recovery using the vpfsd service 674

Chapter 16 Configuring isolated recovery environment (IRE) 675

Requirements 675
Configuring the network isolation 676
Configuring an isolated recovery environment using the web UI 678
Configuring the allowed subnets 679

Configuring the reverse connections 679
Configuring the reverse replication schedule 681
Adding a replication operation to SLP at the production primary server 681
Configuring an isolated recovery environment using the command line 683
Using the API Keys to support MFA configuration 683
Configuring an isolated recovery environment on a NetBackup BYO media server 684
Managing an isolated recovery environment on a NetBackup BYO media server 689
Configuring A.I.R. for replicating backup images from production environment to IRE BYO environment 692
Configuring an isolated recovery environment on a WORM storage server 696
Managing an isolated recovery environment on a WORM storage server 699
Configuring data transmission between a production environment and an IRE WORM storage server 702
Replicating the backup images from the IRE domain to the production domain 705

Chapter 17 Using the NetBackup Deduplication Shell 708

About the NetBackup Deduplication Shell 709
Managing users from the deduplication shell 710
Adding and removing local users from the deduplication shell 710
Adding MSDP users from the deduplication shell 711
Adding MSDP admin alias users from the deduplication shell 712
Connecting an Active Directory domain to a WORM or an MSDP storage server for Universal Shares and Instant Access 713
Disconnecting an Active Directory domain from the deduplication shell 714
Changing a user password from the deduplication shell 714
Managing VLAN interfaces from the deduplication shell 716
Managing the retention policy on a WORM storage server 717
Managing images with a retention lock on a WORM storage server 718
Auditing WORM retention changes 719
Protecting the MSDP catalog from the deduplication shell 720
About the external MSDP catalog backup 721

Configuring an external MSDP catalog backup from the deduplication shell 721
Restoring from the external MSDP catalog backup 723
Troubleshooting the external MSDP catalog backup 723
Managing certificates from the deduplication shell 724
Viewing the certificate details from the deduplication shell 724
Importing certificates from the deduplication shell 725
Removing certificates from the deduplication shell 727
Managing FIPS mode from the deduplication shell 728
Managing post-quantum cryptography (PQC) mode from the deduplication shell 729
Encrypting backups from the deduplication shell 730
Tuning the MSDP configuration from the deduplication shell 731
Setting the MSDP log level from the deduplication shell 736
Managing NetBackup services from the deduplication shell 737
Managing the cyclic redundancy checking (CRC) service 738
Managing the content router queue processing (CRQP) service 739
Managing the online checking service 739
Managing the compaction service 740
Managing the deduplication (MSDP) services 740
Managing the MSDP services across the cluster 741
Managing the Storage Platform Web Service (SPWS) 742
Managing Open Cloud Storage Daemon 743
Managing the Veritas provisioning file system (VPFS) configuration parameters 744
Managing the Veritas provisioning file system (VPFS) mounts 745
Managing the NGINX service 746
Managing the SMB service 747
Monitoring and troubleshooting NetBackup services from the deduplication shell 748
Managing the health monitor 749
Viewing information about the system 749
Viewing the deduplication (MSDP) history or configuration files 750
Viewing process information in the pseudo-file system 751
Viewing the deduplication rate of a Veritas provisioning file service (VPFS) share 752
Viewing the log files 752
Collecting and transferring troubleshooting files 754
Managing S3 service from the deduplication shell 755
Configuring the S3 service 756

Creating or resetting root credentials 756
Changing the S3 service certificates 756
Managing the S3 service 757
Changing S3 service log level 757
Multi-person authorization for deduplication shell commands 757
Managing cloud LSU in Flex Scale and Cloud Scale 758
Managing the NFS version 3 server services for the MSDP container 759
Viewing the NetBackup RBAC roles assigned to the MSDP container 759

Chapter 18 Troubleshooting 761

About unified logging 761
About using the `vxlogview` command to view unified logs 762
Examples of using vxlogview to view unified logs 764
About legacy logging 766
Creating NetBackup log file directories for MSDP 766
NetBackup MSDP log files 767
Troubleshooting MSDP configuration issues 773
MSDP storage server configuration fails 773
MSDP database system error (220) 774
MSDP server not found error 774
License information failure during MSDP configuration 775
The disk pool tab does not display an MSDP volume 775
Troubleshooting MSDP operational issues 776
Verify that the MSDP server has sufficient memory 776
MSDP backup or duplication job fails 776
MSDP client deduplication fails 779
MSDP volume state changes to DOWN when volume is unmounted 779
MSDP errors, delayed response, hangs 780
Cannot delete an MSDP disk pool 781
MSDP media open error (83) 782
MSDP media write error (84) 784
MSDP no images successfully processed (191) 785
MSDP storage full conditions 786
Troubleshooting MSDP catalog backup 786
Storage Platform Web Service (spws) does not start 787
Disk volume API or command line option does not work 787
Viewing MSDP disk errors and events 788
MSDP event codes and messages 788

Unable to obtain the administrator password to use an AWS EC2 instance that has a Windows OS 791
Trouble shooting multi-domain issues 791
Unable to configure OpenStorage server from another domain 791
MSDP storage server is down when you configure an OpenStorage server 792
MSDP server is overloaded when it is used by multiple NetBackup domains 793
Troubleshooting the cloud compaction error messages 794
Troubleshooting the msdpcmdrun issues 794

Appendix A Migrating to MSDP storage 797

Migrating from another storage type to MSDP 797

Appendix B Migrating from Cloud Catalyst to MSDP direct cloud tiering 799

About migration from Cloud Catalyst to MSDP direct cloud tiering 799
About Cloud Catalyst migration strategies 800
About direct migration from Cloud Catalyst to MSDP direct cloud tiering 805
About requirements for a new MSDP direct cloud tier storage server 805
About beginning the direct migration 806
Placing the Cloud Catalyst server in a consistent state 807
About installing and configuring the new MSDP direct cloud tier server 809
Running the migration to the new MSDP direct cloud tier server 810
About postmigration configuration and cleanup 816
About the Cloud Catalyst migration `-dryrun` option 818
About Cloud Catalyst migration `cacontrol` options 818
Reverting back to Cloud Catalyst from a successful migration 821
Reverting back to Cloud Catalyst from a failed migration 823

Appendix C Encryption Crawler 826

About the Encryption Crawler 826
About the two modes of the Encryption Crawler 827
Managing the Encryption Crawler 829
Advanced options 835
Tuning options 836

Encrypting the data 839
Command usage example outputs 840
Updating the KMS configuration 846
Converting the legacy KMS to KEK-based KMS 846
Performing the KMS key rotation 847
Performing the KEK rotation 847
Migrating the KMS vendor 848

Index 850

Chapter 1

Introducing the NetBackup media server deduplication option

This chapter includes the following topics:

- About the NetBackup deduplication options

About the NetBackup deduplication options

Cohesity NetBackup provides the deduplication options that let you deduplicate data everywhere, as close to the source of data as you require.

Deduplication everywhere provides the following benefits:

- Reduce the amount of data that is stored.
- Reduce backup bandwidth.
- Reduce backup windows.
- Reduce infrastructure.

Deduplication everywhere lets you choose at which point in the backup process to perform deduplication. NetBackup can manage your deduplication wherever you implement it in the backup stream.

Table 1-1 describes the options for deduplication.

Table 1-1 NetBackup deduplication options

Type	Description
Media server deduplication	NetBackup clients send their backups to a NetBackup media server, which deduplicates the backup data. A NetBackup media server hosts the NetBackup Deduplication Engine, which writes the data to a **Media Server Deduplication Pool** on the target storage and manages the deduplicated data See "About NetBackup media server deduplication" on page 34.
Client deduplication	With NetBackup MSDP client deduplication, clients deduplicate their backup data and then send it directly to the storage server, which writes it to the storage. The network traffic is reduced greatly. See "About NetBackup Client Direct deduplication" on page 38.
NetBackup appliance deduplication	Cohesity provides several hardware and a software solutions that include NetBackup deduplication. The NetBackup appliances have their own documentation set: https://sort.veritas.com/documents

Chapter 2

Planning your deployment

This chapter includes the following topics:

- Planning your MSDP deployment
- NetBackup naming conventions
- About MSDP deduplication nodes
- About the NetBackup deduplication destination
- About MSDP capacity support and hardware requirements
- About MSDP storage and connectivity requirements
- About NetBackup media server deduplication
- About NetBackup Client Direct deduplication
- About MSDP remote office client deduplication
- About the NetBackup Deduplication Engine credentials
- About the network interface for MSDP
- About MSDP port usage
- About MSDP optimized synthetic backups
- About MSDP and SAN Client
- About MSDP optimized duplication and replication
- About MSDP stream handlers
- MSDP deployment best practices

Planning your MSDP deployment

Table 2-1 provides an overview of planning your deployment of NetBackup deduplication.

Table 2-1 Deployment overview

Step	Deployment task	Where to find the information
Step 1	Learn about deduplication nodes and storage destinations	See "About MSDP deduplication nodes" on page 27. See "About the NetBackup deduplication destination" on page 27.
Step 2	Understand the storage capacity and requirements	See "About MSDP capacity support and hardware requirements" on page 28. See "About MSDP storage and connectivity requirements" on page 30.
Step 3	Determine which type of deduplication to use	See "About NetBackup media server deduplication" on page 34. See "About NetBackup Client Direct deduplication" on page 38. See "About MSDP remote office client deduplication" on page 40.
Step 4	Determine the requirements for deduplication hosts	See "About MSDP storage servers" on page 35. See "About MSDP server requirements" on page 37. See "About MSDP client deduplication requirements and limitations" on page 40. See "About the network interface for MSDP" on page 42. See "About MSDP port usage" on page 43. See "About scaling MSDP" on page 51.
Step 5	Determine the credentials for deduplication	See "About the NetBackup Deduplication Engine credentials" on page 41.
Step 6	Read about compression and encryption	See "About MSDP compression" on page 108. See "About MSDP encryption" on page 109.
Step 7	Read about optimized synthetic backups	See "About MSDP optimized synthetic backups" on page 44.
Step 8	Read about deduplication and SAN Client	See "About MSDP and SAN Client" on page 45.
Step 9	Read about optimized duplication and replication	See "About MSDP optimized duplication and replication" on page 45.
Step 10	Read about stream handlers	See "About MSDP stream handlers" on page 46.

Table 2-1 Deployment overview *(continued)*

Step	Deployment task	Where to find the information
Step 11	Read about best practices for implementation	See "MSDP deployment best practices" on page 50.
Step 12	Determine the storage requirements and provision the storage	See "About provisioning the storage for MSDP" on page 57. See "About MSDP storage and connectivity requirements" on page 30. See "About MSDP capacity support and hardware requirements" on page 28. See "MSDP storage path properties" on page 95.
Step 13	Configure MSDP	See "Configuring media server deduplication in NetBackup" on page 63. See "Configuring MSDP client-side deduplication" on page 65.
Step 14	Migrate from other storage to NetBackup deduplication	See "Migrating from another storage type to MSDP" on page 797.

NetBackup naming conventions

NetBackup has rules for naming logical constructs, such as clients, disk pools, backup policies, storage lifecycle policies, and so on. Generally, names are case-sensitive. The following set of characters can be used in user-defined names and passwords:

- Alphabetic (A-Z a-z) (names are case-sensitive)
- Numeric (0-9)
- Period (.)
  Do not use periods in the WORM volume names.
- Plus (+)
- Hyphen (-)
  Do not use a hyphen as the first character.
- Underscore (_)

These characters are also used for foreign languages.

Note: No spaces are allowed.

The Logical Storage Unit (LSU) name or the Domain Volume name must have fewer than 50 ASCII characters including a hyphen (-) and an underscore (_) and must not have a blank space.

The naming conventions for the NetBackup Deduplication Engine differ from these NetBackup naming conventions.

See "About the NetBackup Deduplication Engine credentials" on page 41.

About MSDP deduplication nodes

A media server deduplication node comprises the following:

Storage server	The storage server deduplicates the backups, writes the data to the storage, and manages the storage. See "About MSDP storage servers" on page 35.
Load balancing servers	Load balancing servers assist the storage server by deduplicating backups. Load balancing servers are optional. See "About MSDP load balancing servers" on page 36.
Storage	See "About the NetBackup deduplication destination" on page 27.
Clients	The clients may include the clients that deduplicate their own data (Client Direct). See "About NetBackup Client Direct deduplication" on page 38.

Multiple media server deduplication nodes can exist. Nodes cannot share servers or storage.

Each node manages its own storage. Deduplication within each node is supported; deduplication between nodes is not supported.

See "About NetBackup media server deduplication" on page 34.

See "About MSDP storage servers" on page 35.

About the NetBackup deduplication destination

Media Server Deduplication Pool

A NetBackup **Media Server Deduplication Pool** represents the disk or cloud storage that is attached to a NetBackup media server. NetBackup deduplicates the data and hosts the storage.

If you use this destination, use this guide to plan, implement, configure, and manage deduplication and the storage. When you configure the storage server, select **Media Server Deduplication Pool** as the storage type.

The **Media Server Deduplication Pool** can be hosted on the following systems:

- A NetBackup media server.
- A NetBackup 5200 series appliance or NetBackup 5300 series appliance.
- A NetBackup Flex Appliance media server instance.

About MSDP capacity support and hardware requirements

The MSDP storage contains one local LSU or multiple cloud LSUs.

NetBackup 10.2 introduced a new deduplication fingerprint lookup cache referred to as Predictive/Sampling (P/S) cache, enabling support for larger MSDP pools. The new MSDP pools that are configured with version 10.2 or later use this P/S cache by default. If you upgrade the existing MSDP pools to the version 10.2, you are on the existing MSDP cache architecture and the MSDP limit does not change.

If you upgrade to NetBackup 10.4, you have the option to convert your existing MSDP pools to the new P/S cache algorithm and increase the supported capacity of your local and cloud storage.

See “About sampling and predictive cache” on page 86.

See “Rebuilding the sampling cache” on page 88.

The following table lists the MSDP capacity for NetBackup 10.1.1 and earlier versions.

Table 2-2 MSDP capacity for NetBackup 10.1.1 and earlier versions

Platform	Local disk pool	Local and cloud disk pool
BYO	400 TiB	1.2 PiB
Cloud-Only	N/A	1.2 PiB
NBA	960 TiB	1.2 PiB

Table 2-2 MSDP capacity for NetBackup 10.1.1 and earlier versions *(continued)*

Platform	Local disk pool	Local and cloud disk pool
Flex	960 TiB	1.2 PiB
Flex Scale (16-node)	1.8 PiB	8.8 PiB
Access	1.2 PiB	N/A

The following table lists the MSDP capacity for NetBackup version 10.2 and later with P/S cache MSDP pools.

Table 2-3 MSDP capacity for NetBackup 10.2 and later

Platform	Local disk pool	Local and cloud disk pool
BYO	400 TiB	2.4 PiB
Cloud-Only	N/A	2.0 PiB
NBA	960 TiB	2.4 PiB
Flex	1.2 PiB 2.4 PiB (5372 appliance with 20-TB drives)	2.4 PiB
Flex Scale (16-node)	2.5 PiB	8.8 PiB
Access	2.4 PiB	N/A
Cloud Scale (16-node)	N/A	4.0 PiB

Note: Increased pool size is supported in Access version 8.4.

Note: Although the new P/S cache enables support of larger MSDP pools, you must ensure that you have the appropriate resources available to support the larger pools.

Note: On a Flex 5372 appliance that is configured with 20-TB disk drives, a local Media Server Deduplication Pool is supported up to 2.4 PiB. This configuration requires Flex Appliance 7.0 or later and the storage shelf configuration that provides the required usable capacity. On all other Flex appliance models and drive capacities, the local pool limit remains 1.2 PiB. For the supported storage shelf and drive configurations, see the *Cohesity 5372 Appliance Product Description Guide*.

To identify the supported applications and usage information for Flex Appliance, see the following article:

https://support.cohesity.com/s/article/article-100042995

NetBackup reserves 4 percent of the storage space for the deduplication database and transaction logs. Therefore, a storage full condition is triggered at a 96-percent threshold. If you use separate storage for the deduplication database, NetBackup still uses the 96-percent threshold to protect the data storage from any possible overload.

If your storage requirements exceed the capacity of a **Media Server Deduplication Pool**, you can use more than one media server deduplication node.

See "About MSDP deduplication nodes" on page 27.

For the operating system versions that NetBackup supports for deduplication, see the NetBackup Software Compatibility List.

About MSDP storage and connectivity requirements

The following subsections describe the storage and the connectivity requirements for the NetBackup Media Server Deduplication Option.

Storage media

The following are the minimum requirements for single stream read or write performance for each disk volume. Greater individual data stream capability or aggregate capability may be required to satisfy your objectives for writing to and reading from disk.

Up to 32 TBs of storage	130 MB/sec. 200 MB/sec for enterprise-level performance.

32 to 48 TBs of storage	200 MB/sec. Cohesity recommends that you store the data and the deduplication database on separate disk volumes, each with 200 MB/sec read or write speed. Neither should be stored on the system disk.
48 to 64 TBs of storage	250 MB/sec. Cohesity recommends that you store the data and the deduplication database on separate disk volumes, each with 250 MB/sec read or write speed. Neither should be stored on the system disk.
96 TBs of storage	250 MB/sec. 96 TBs of storage require four separate volumes, each with 250 MB/sec read or write speed. You cannot use the system disk of the storage server host for any of the required volumes.
400 TBs of storage	500 MB/sec

Local disk storage may leave you vulnerable in a disaster. SAN disk can be remounted at a newly provisioned server with the same name.

When you deploy NetBackup, provide a dedicated file system for the MSDP storage. If the file system is used for MSDP storage is shared with other applications, it may result in a performance degradation, and affect the reporting of storage utilization. If another application writes an excessive amount of data, the file system may get full unexpectedly. If storage reaches 96% of the capacity, the MSDP storage server becomes unavailable for the backup jobs.

NetBackup **Media Server Deduplication Pool** does not support the following storage types for deduplication storage:

- Network Attached Storage (that is, file based storage protocols) such as CIFS or NFS.
- The ZFS file system.

The NetBackup compatibility lists are the definitive source for supported operating systems, computers, and peripherals. See the compatibility lists available at the following website:

https://support.cohesity.com/s/article/article-100040093

The storage must be provisioned and operational before you can configure deduplication in NetBackup.

See “About provisioning the storage for MSDP” on page 57.

Storage connection

The storage must be direct-attached storage (DAS), internal disks, or connected by a dedicated, low latency storage area network (Fibre Channel or iSCSI).

A storage area network should conform to the following criteria:

Latency	Maximum 0.1-millisecond latency per round trip.
Bandwidth	Enough bandwidth on the storage network to satisfy your throughput objectives. Cohesity supports iSCSI on storage networks with at least 10-Gigabit Ethernet network bandwidth. Cohesity recommends the Fibre Channel storage networks with at least 4-Gigabit network bandwidth.
HBAs	The storage server should have an HBA or HBAs dedicated to the storage. Those HBAs must have enough bandwidth to satisfy your throughput objectives.

See “Fibre Channel and iSCSI comparison for MSDP” on page 32.

See “About NetBackup media server deduplication” on page 34.

See “MSDP storage path properties” on page 95.

See “Planning your MSDP deployment” on page 25.

See “About MSDP capacity support and hardware requirements” on page 28.

Fibre Channel and iSCSI comparison for MSDP

Deduplication is a CPU and memory intensive process. It also requires dedicated and high-speed storage connectivity for the best performance. That connectivity helps to ensure the following:

- Consistent storage performance.
- Reduced packet loss during network congestion.
- Reduced storage deadlocks.

The following table compares both the Fibre Channel and the iSCSI characteristics that affect deduplication storage performance. By design, Fibre Channel provides the greatest opportunity to meet performance objectives. To achieve the results that are required for NetBackup MSDP storage, iSCSI may require other optimizations that are described in the following table.

Table 2-4 Fibre Channel and iSCSI characteristics

Item	Fibre Channel	iSCSI
Genesis	Storage networking architecture that is designed to handle the same block storage format that storage devices use.	Storage network protocol that is built on top of TCP/IP to use the same wiring as the rest of the enterprise.
Protocol	FCP is a thin, single-purpose protocol that provides lossless, in-order frame delivery and low switch latency.	iSCSI is a multiple layer implementation that facilitates data transfers over intranets and long distances. The SCSI protocol expects lossless, in-order delivery, but iSCSI uses TCP/IP, which experiences packet loss and out-of-order delivery.
Host CPU load	Low. Fibre Channel frame processing is offloaded to dedicated low-latency HBAs.	Higher. Most iSCSI implementations use the host processor to create, send, and interpret storage commands. Therefore, Cohesity requires dedicated network interfaces on the storage server to reduce storage server load and reduce latency.
Latency	Low.	Higher.
Flow control	A built-in flow control mechanism that ensures data is sent to a device when it is ready to accept it.	No built-in flow control. Cohesity recommends that you use the Ethernet priority-based flow control as defined in the IEEE 802.1Qbb standard.
Deployment	Difficult.	Easier than Fibre Channel, but more difficult to deploy to meet the criteria for MSDP. The required dedicated network interfaces add to deployment difficult. Other optimizations for carrying storage traffic also add to deployment difficult. Other optimizations include flow control, jumbo framing, and multi-path I/O.

Although Cohesity supports iSCSI for connectivity to **Media Server Deduplication Pool** storage, Cohesity recommends Fibre Channel. Cohesity believes that Fibre Channel provides better performance and stability than iSCSI. iSCSI instability may manifest as status 83 and status 84 error messages.

See “MSDP media open error (83)” on page 782.

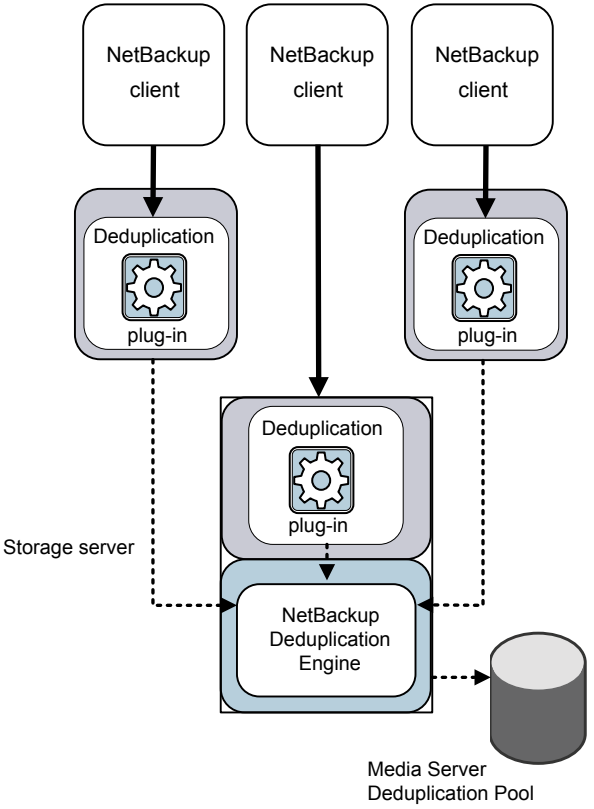
See “MSDP media write error (84)” on page 784.

About NetBackup media server deduplication

With media server deduplication, the NetBackup client software creates the image of backed up files as for a normal backup. The client sends the backup image to a media server, which hosts the plug-in that duplicates the backup data. The media server can be the storage server or a load balancing server if one is configured. The deduplication plug-in breaks the backup image into segments and compares the segments to all of the segments that are stored in that deduplication node. The plug-in then sends only the unique segments to the NetBackup Deduplication Engine on the storage server. The Deduplication Engine writes the data to a **Media Server Deduplication Pool**.

Figure 2-1 shows NetBackup media server deduplication. The deduplication storage server is a media server on which the deduplication core components are enabled. The storage destination is a **Media Server Deduplication Pool**.

Figure 2-1 NetBackup media server deduplication



More detailed information is available.

See "About MSDP deduplication nodes" on page 27.

See "About MSDP storage servers" on page 35.

See "About MSDP load balancing servers" on page 36.

See "About MSDP server requirements" on page 37.

See "About MSDP unsupported configurations" on page 38.

See "MSDP server components" on page 593.

See "Media server deduplication backup process" on page 596.

About MSDP storage servers

A storage server is an entity that writes to and reads from the storage. One host functions as the storage server, and only one storage server exists for each NetBackup deduplication node. The host must be a NetBackup media server. Although the storage server components run on a media server, the storage server is a separate logical entity.

See "About MSDP deduplication nodes" on page 27.

The MSDP storage server does the following:

- Receives the backups from clients and then deduplicates the data.
- Receives the deduplicated data from clients or from other media servers.
  You can configure NetBackup clients and other NetBackup media servers to deduplicate data also. In which case, the storage server only receives the data after it is deduplicated.
  See "About NetBackup Client Direct deduplication" on page 38.
  See "About MSDP load balancing servers" on page 36.
- Writes the deduplicated data to and reads the deduplicated data from the disk or cloud storage.
- Manages that storage.
- Manages the deduplication processes.

How many storage servers (and by extension, nodes) you configure depends on your storage requirements. It also depends on whether or not you use optimized duplication or replication, as follows:

- Optimized duplication between local LSUs in the same domain requires at least two deduplication nodes in the same domain. The following are the required storage servers:
  - One for the backup storage, which is the source for the duplication operations.

- Another to store the copies of the backup images, which are the target for the duplication operations.

See "About MSDP optimized duplication within the same domain" on page 122.

- Auto Image Replication to another domain requires the following storage servers:
  - One for the backups in the originating NetBackup domain. This storage server writes the NetBackup client backups to the storage. It is the source for the duplication operations.
  - Another in the remote NetBackup domain for the copies of the backup images. This storage server is the target for the duplication operations that run in the originating domain.

  See "About NetBackup Auto Image Replication" on page 139.

About MSDP load balancing servers

You can configure other NetBackup media servers to help deduplicate data. They perform file fingerprint calculations for deduplication, and they send the unique data segments to the storage server. These helper media servers are called load balancing servers.

A NetBackup media server becomes a load balancing server when two things occur:

- You enable the media server for deduplication load balancing duties.
  You do so when you configure the storage server or later by modifying the storage server properties.
- You select it in the storage unit for the deduplication pool.

See "Introduce MSDP load balancing servers gradually" on page 52.

Load balancing servers also perform restore and duplication jobs.

Load balancing servers can be any supported server type for deduplication. They do not have to be the same type as the storage server.

Note: If you want to use Fibre Channel for VMware backups in a Cohesity appliance configuration, the same Fibre Channel datastore LUNs must be zoned to all of the load balancing media servers.

See "About MSDP storage servers" on page 35.

See "About NetBackup media server deduplication" on page 34.

See "About MSDP storage servers" on page 35.

See "Managing MSDP servers" on page 531.

About MSDP server requirements

The host computer's CPU and memory constrain how many jobs can run concurrently. The storage server requires enough capability for deduplication and for storage management unless you offload some of the deduplication to load-balancing servers.

Table 2-5 shows the minimum requirements for MSDP servers. NetBackup deduplication servers are always NetBackup media servers.

Processors for deduplication should have a high clock rate and high floating point performance. Furthermore, high throughput per core is desirable. Each backup stream uses a separate core.

Intel and AMD have similar performance and perform well on single core throughput.

Newer SPARC processors, such as the SPARC64 VII, provide the single core throughput that is similar to AMD and Intel. Alternatively, UltraSPARC T1 and T2 single core performance does not approach that of the AMD and Intel processors. Tests show that the UltraSPARC processors can achieve high aggregate throughput. However, they require eight times as many backup streams as AMD and Intel processors to do so.

Table 2-5 MSDP server minimum requirements

Component	Storage server	Load-balancing server
CPU	Cohesity recommends at least a 2.2-GHz clock rate. A 64-bit processor is required. At least four cores are required. Cohesity recommends eight cores. For 64 TBs of storage, Intel x86-64 architecture requires eight cores.	Cohesity recommends at least a 2.2-GHz clock rate. A 64-bit processor is required. At least two cores are required. Depending on throughput requirements, more cores may be helpful.
RAM	From 8 TBs to 32 TBs of storage, Cohesity recommends 1GB RAM for 1TB of storage. However if you go beyond 32 TBs of storage, Cohesity recommends more than 32GBs of RAM for better and enhanced performance.	4 GBs.
Operating system	The operating system must be a supported 64-bit operating system. See the NetBackup Software Compatibility List for your NetBackup release on the Cohesity Support website.	The operating system must be a supported 64-bit operating system. See the NetBackup Software Compatibility List for your NetBackup release on the following website.

Note: In some environments, a single host can function as both a NetBackup primary server and as a deduplication server. Such environments typically run fewer than 100 total backup jobs a day. (Total backup jobs are backups to any storage destination, including deduplication and non-deduplication storage.) If you perform more than 100 backups a day, deduplication operations may affect primary server operations.

See "About MSDP queue processing" on page 552.

See "About NetBackup media server deduplication" on page 34.

See "Managing MSDP servers" on page 531.

About MSDP unsupported configurations

The following items describe some configurations that are not supported:

- NetBackup media server deduplication and Arctera Backup Exec deduplication cannot reside on the same host. If you use both NetBackup and Backup Exec deduplication, each product must reside on a separate host.
- NetBackup does not support clustering of deduplication storage servers or load balancing servers.
- Deduplication within each media server deduplication node is supported; global deduplication between nodes is not supported.

About NetBackup Client Direct deduplication

With NetBackup Client Direct deduplication (also known as *client-side deduplication*), the client hosts the plug-in that duplicates the backup data. The NetBackup client software creates the image of backed up files as for a normal backup. Next, the deduplication plug-in breaks the backup image into segments and compares the segments to all of the segments that are stored in that deduplication node. The plug-in then sends only the unique segments to the NetBackup Deduplication Engine on the storage server. The engine writes the data to a **Media Server Deduplication Pool**.

Client deduplication does the following:

- Reduces network traffic. The client sends only unique file segments to the storage server. Duplicate data is not sent over the network.
- Distributes some deduplication processing load from the storage server to clients. (NetBackup does not balance load between clients; each client deduplicates its own data.)

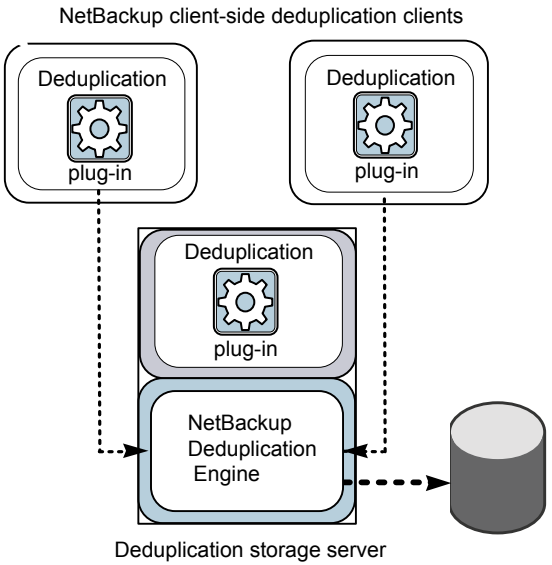
NetBackup Client Deduplication is a solution for the following cases:

- Remote office or branch office backups to the data center.
  NetBackup provides resilient network connections for remote office backups.
  See "About MSDP remote office client deduplication" on page 40.
- LAN connected file server
- Virtual machine backups.

Client-side deduplication is also a useful solution if a client host has unused CPU cycles or if the storage server or load balancing servers are overloaded.

Figure 2-2 shows client deduplication. The deduplication storage server is a media server on which the deduplication core components are enabled. The storage destination is a **Media Server Deduplication Pool**

Figure 2-2 NetBackup client deduplication



More information is available.

See "About MSDP client deduplication requirements and limitations" on page 40.

See "About MSDP remote office client deduplication" on page 40.

See "MSDP client components" on page 597.

See "MSDP client–side deduplication backup process" on page 598.

About MSDP client deduplication requirements and limitations

NetBackup does not support the following for client-side deduplication:

- Multiple copies per job. For the jobs that specify multiple copies, the backup images are sent to the storage server and may be deduplicated there. Multiple copies are configured in a NetBackup backup policy.
- NDMP hosts. The backup jobs fail if you try to use client-side deduplication for NDMP hosts.

For the systems on which NetBackup supports client-side deduplication, see the NetBackup Software Compatibility List.

The clients that deduplicate their own data conform to the standard NetBackup release level compatibility. The NetBackup Release Notes for each release defines the compatibility between NetBackup releases. To take advantage of any new features, improvements, and fixes, Cohesity recommends that the clients and the servers be at the same release and revision.

See "About NetBackup Client Direct deduplication" on page 38.

See "About the NetBackup deduplication options" on page 22.

See "About MSDP server requirements" on page 37.

About MSDP remote office client deduplication

WAN backups require more time than local backups in your own domain. WAN backups have an increased risk of failure when compared to local backups. To help facilitate WAN backups, NetBackup provides the capability for resilient network connections. A resilient connection allows backup and restore traffic between a client and NetBackup media servers to function effectively in high-latency, low-bandwidth networks such as WANs.

The use case that benefits the most from resilient connections is client-side deduplication at a remote office that does not have local backup storage. The following items describe the advantages:

- Client deduplication reduces the time that is required for WAN backups by reducing the amount of data that must be transferred.
- The resilient connections provide automatic recovery from network failures and latency (within the parameters from which NetBackup can recover).

When you configure a resilient connection, NetBackup uses that connection for the backups. Use the NetBackup **Resilient Network** host properties to configure NetBackup to use resilient network connections.

See "Resilient network properties" on page 170.

See “Specify resilient connections for clients” on page 172.

The `pd.confFILE_KEEP_ALIVE_INTERVAL` parameter lets you configure the frequency of keep-alive operations on idle sockets.

See “MSDP pd.conf file parameters” on page 179.

You can improve the performance of the first backup for a remote client.

See “About seeding the MSDP fingerprint cache for remote client deduplication” on page 81.

About MSDP remote client data security

NetBackup supports encryption of data-in-transit to encrypt backup/restore data in connection traffic. See *Encryption of data-in-transit* in *NetBackup Security and Encryption Guide*.

If data-in-transit encryption (DTE) is not enabled, the NetBackup deduplication process can encrypt the data before it is transmitted over the WAN.

See “About MSDP encryption” on page 109.

MSDP supports client direct restore. If the backup data is encrypted, the encrypted data is transferred to the client and is decrypted there.

For client direct restore, See “ Configuring MSDP restores directly to a client” on page 565.

About remote client backup scheduling

NetBackup backup policies use the time zone of the primary server for scheduling jobs. If your remote clients are in a different time zone than your NetBackup primary server, you must compensate for the difference. For example, suppose the primary server is in Finland (UTC+2) and the remote client is in London (UTC+0). If the backup policy has a window from 6pm to 6am, backups can begin at 4pm on the client. To compensate, you should set the backup window from 8pm to 8am. Alternatively, it may be advisable to use a separate backup policy for each time zone in which remote clients reside.

About the NetBackup Deduplication Engine credentials

The NetBackup Deduplication Engine requires credentials. The deduplication components use the credentials when they communicate with the NetBackup

Deduplication Engine. The credentials are for the deduplication engine, not for the host on which it runs.

You enter the NetBackup Deduplication Engine credentials when you configure the storage server.

The following are the rules for the credentials:

- The user name and the password can be up to 62 characters in length. The user name and the password cannot be empty.
- You can use characters in the printable ASCII range (0x20-0x7E) except for the following characters:
  - Asterisk (*)
  - Backward slash (\) and forward slash (/)
  - Double quote (")
  - Left parenthesis [(] and right parenthesis [)]
  - Less than (<) and greater than (>) sign.
  - Caret sign (^).
  - Percent sign (%).
  - Ampersand (&)
  - Spaces.
  - Leading and trailing quotes.
  - Square brackets ([])
  - At sign (@)

Cohesity appliance products that use deduplication engine may have more restrictive password requirements than mentioned here. See the appliance-specific documentation for password guidelines.

Note: You cannot change the NetBackup Deduplication Engine credentials after you enter them. Therefore, carefully choose and enter your credentials. If you must change the credentials, contact your Cohesity support representative.

About the network interface for MSDP

If the MSDP storage server has more than one network interface, NetBackup uses the default interface for all deduplication traffic. (Deduplication traffic includes backups, restores, duplication, and replication.) The host operating system

determines which network interface is the default. However, you can configure the network interface or interfaces that NetBackup uses, as follows:

Configure a specific interface	To use a specific interface, you can enter that interface name when you configure the deduplication storage server. NetBackup uses this interface for all deduplication traffic unless you also configure a separate interface for duplication and replication. See "MSDP network interface properties" on page 97. See "Configuring a storage server for a Media Server Deduplication Pool" on page 93.
Configure an interface for duplication and replication traffic	You can configure a separate network interface for the duplication and the replication traffic. The backup and restore traffic continues to use the default interface or the specific configured interface. See "About a separate network path for MSDP duplication and replication" on page 120. See "Configuring a separate network path for MSDP duplication and replication" on page 121.

The NetBackup `REQUIRED_INTERFACE` setting does not affect deduplication processes.

See "About NetBackup media server deduplication" on page 34.

See "Planning your MSDP deployment" on page 25.

See "Configuring a storage server for a Media Server Deduplication Pool" on page 93.

About MSDP port usage

The following table shows the ports that are used for NetBackup deduplication. If firewalls exist between the various deduplication hosts, open the indicated ports on the deduplication hosts. Deduplication hosts are the deduplication storage server, the load balancing servers, and the clients that deduplicate their own data.

If you have only a storage server and no load balancing servers or clients that deduplicate their own data: you do not have to open firewall ports.

Table 2-6 Deduplication ports

Port	Usage
10082	The NetBackup Deduplication Engine (`spoold`). Open this port between the hosts that deduplicate data. Hosts include load balancing servers and the clients that deduplicate their own data.
10102	The NetBackup Deduplication Manager (`spad`). Open this port between the hosts that deduplicate data. Hosts include load balancing servers and the clients that deduplicate their own data.
443	Open this port between the MSDP server and the cloud storage target such as AWS or Azure.

About MSDP optimized synthetic backups

Optimized synthetic backups are a more efficient form of synthetic backup. A media server uses messages to instruct the storage server which full and incremental backup images to use to create the synthetic backup. The storage server constructs (or synthesizes) the backup image directly on the disk storage. Optimized synthetic backups require no data movement across the network.

The optimized synthetic backup method provides the following benefits:

- Faster than a synthetic backup.
  Regular synthetic backups are constructed on the media server. They are moved across the network from the storage server to the media server and synthesized into one image. The synthetic image is then moved back to the storage server.
- Requires no data movement across the network.
  Regular synthetic backups use network traffic.

See “Configuring optimized synthetic backups for MSDP” on page 120.

In NetBackup, the **OptimizedImage** attribute enables optimized synthetic backups. It applies to both storage servers and deduplication pools. This attribute is enabled by default on storage servers and media server deduplication pools.

See “Setting MSDP storage server attributes” on page 533.

See “Setting a Media Server Deduplication Pool attribute” on page 543.

Table 2-7 MSDP requirements and limitations for optimized synthetic backups

What	Description
Requirements	The target storage unit's deduplication pool must be the same deduplication pool on which the source images reside.
Limitations	NetBackup does not support storage unit groups as a destination for optimized synthetic backups. If NetBackup cannot produce the optimized synthetic backup, NetBackup creates the more data-movement intensive synthetic backup.

About MSDP and SAN Client

SAN Client is a NetBackup optional feature that provides high speed backups and restores of NetBackup clients. Fibre Transport is the name of the NetBackup high-speed data transport method that is part of the SAN Client feature. The backup and restore traffic occurs over a SAN.

SAN clients can be used with the deduplication option; however, the deduplication must occur on the media server, not the client. Configure the media server to be both a deduplication storage server (or load balancing server) and an FT media server. The SAN client backups are then sent over the SAN to the deduplication server/FT media server host. At that media server, the backup stream is deduplicated.

Do not enable client-side deduplication on SAN Clients. The data processing for deduplication is incompatible with the high-speed transport method of Fibre Transport. Client-side deduplication relies on two-way communication over the LAN with the media server. A SAN client streams the data to the FT media server at a high rate over the SAN.

About MSDP optimized duplication and replication

NetBackup supports several methods for optimized duplication and replication of deduplicated data.

The following table lists the duplication methods NetBackup supports between media server deduplication pools.

Table 2-8 NetBackup OpenStorage optimized duplication and replication methods

Optimized duplication method	Description
Within the same NetBackup domain	See "About MSDP optimized duplication within the same domain" on page 122. See "About MSDP cloud support" on page 294.
To a remote NetBackup domain	See "About NetBackup Auto Image Replication" on page 139.

About MSDP stream handlers

NetBackup provides the stream handlers that process various backup data stream types. Stream handlers improve backup deduplication rates by processing the underlying data stream.

For data that has already been deduplicated, the first backup with a new stream handler produces a lower deduplication rate. After that first backup, the deduplication rate should surpass the rate from before the new stream handler was used.

Cohesity continues to develop additional stream handlers to improve backup deduplication performance.

Oracle stream handler

In NetBackup 10.0 and later, the Oracle stream handler is enabled (by default) for all new clients that have no existing images. As with previous versions, the Oracle stream handler only supports stream-based backups and you can configure the Oracle stream handler using the `cacontrol` command line utility. You can enable and disable the stream handler per the following:

- Policy and client
- Policy level
- Stream type level

Note: When you use the Oracle stream handler, it is not recommended to use variable-length deduplication.

The `cacontrol` command utility with the `--sth` flag, is used to override the default behavior of NetBackup by creating a `Marker Entry` for a client, policy, or stream

type in a configuration file. The `cacontrol` command utility is located in the following locations:

- Windows: *install_path*\Cohesity\pdde\cacontrol
- UNIX: /usr/openv/pdde/pdcr/bin/cacontrol

In the following examples for `cacontrol`, *STHTYPE* must be set to `Oracle` to configure the Oracle stream handler.

When using the `cacontrol` command utility to create a `Marker Entry` in NetBackup 10.0 and later, priority is given to the more granular configuration. For example:

```
Marker Entry 1: <Client1> <Policy1> to enabled

Marker Entry 2: <Policy1> to disabled
```

The stream handler is enabled because the more granular configuration in `Marker Entry 1` has higher priority.

In NetBackup 10.0 and later, you can configure `cacontrol` using the following options:

- You can query the settings for the stream handler per client and policy.

  ```
  cacontrol --sth get <STHTYPE> <CLIENT> <POLICY> [SPAUSER]
  ```

- You can enable the stream handler per client and policy.

  ```
  cacontrol --sth update
  <STHTYPE> <CLIENT> <POLICY> [SPAUSER] <enabled>
  ```

- You can delete the settings for a client and policy (return to default behavior).

  ```
  cacontrol --sth delete <STHTYPE> <CLIENT> <POLICY> [SPAUSER]
  ```

- You can disable the stream handler on a client and policy.

  ```
  cacontrol --sth update
  <STHTYPE> <CLIENT> <POLICY> [SPAUSER] <disabled>
  ```

- You can query the settings for the stream handler per policy.

  ```
  cacontrol --sth getbypolicy <STHTYPE> <POLICY> [SPAUSER]
  ```

- You can enable the stream handler per policy.

  ```
  cacontrol --sth updatebypolicy
  <STHTYPE> <POLICY> [SPAUSER] <enabled>
  ```

- You can delete the settings for the stream handler per policy (return to default behavior).

  ```
  cacontrol --sth deletebypolicy <STHTYPE> <POLICY> [SPAUSER]
  ```

- You can disable the stream handler per policy.

  ```
  cacontrol --sth updatebypolicy
  <STHTYPE> <POLICY> [SPAUSER] <disabled>
  ```

- You can query the settings for the stream handler per stream handler type.

  ```
  cacontrol --sth getbytype <STHTYPE> [SPAUSER]
  ```

- You can enable a stream handler per stream handler type.

  ```
  cacontrol --sth updatebytype <STHTYPE> [SPAUSER] <enabled>
  ```

- You can delete the settings for a stream handler (return to default behavior).

  ```
  cacontrol --sth deletebytype <STHTYPE> [SPAUSER]
  ```

- You can disable the stream handler per stream handler type.

  ```
  cacontrol --sth updatebytype <STHTYPE> [SPAUSER] <disabled>
  ```

Microsoft SQL Server stream handler

Microsoft SQL stream handler improves Microsoft SQL backup deduplication rate. You can apply the Microsoft SQL Server stream handler to all the Microsoft SQL Server versions and Azure SQL Server.

You can protect the Microsoft SQL database with NetBackup using the following methods:

- **MS-SQL-Server** policy
  Use **MS-SQL-Server** policy type to protect Microsoft SQL database. This policy type is the recommended way to protect the Microsoft SQL database.
  In this case, the Microsoft SQL stream handler is enabled automatically. The policy has multiple options including a use of a user-provided batch file and intelligent policy.
- **Standard** policy
  Use a standard policy to protect Microsoft SQL dump files.

In this case, the user dumps Microsoft SQL Server to a file and creates a standard policy to back up the dumped file.

You must use `cacontrol` command to enable the Microsoft SQL stream handler manually.

Following are the `cacontrol` options that you can use to manage Microsoft SQL Server stream handler:

Table 2-9 The `cacontrol` command options

Option	Description
`cacontrol --sth get <Oracle|MSSQL> <client> <policy>`	Get the specified marker entry status.
`cacontrol --sth delete <Oracle|MSSQL> <client> <policy>`	Delete the specified marker entry.
`cacontrol --sth update <Oracle|MSSQL> <client> <policy> <enabled|disabled>`	Update the specified marker entry status.
`cacontrol --sth getbypolicy <Oracle|MSSQL> <policy>`	Get the specified marker entry status for a given policy.
`cacontrol --sth deletebypolicy <Oracle|MSSQL> <policy>`	Delete the specified marker entry for a given policy.
`cacontrol --sth updatebypolicy <Oracle|MSSQL> <policy> enabled|disabled`	Update the specified marker entry status for a given policy.

To use the Microsoft SQL Server stream handler

1. Enable or disable the stream handler for a policy.

   ```
   cacontrol --sth updatebypolicy MSSQL <POLICY name>
   enabled/disabled
   ```

2. Delete the stream handler settings for a policy (returns to the default behavior). Microsoft SQL stream handler is enabled by default.

   ```
   cacontrol --sth deletebypolicy MSSQL <POLICY name>
   ```

3. Query the settings for the stream handler for a policy.

   ```
   cacontrol --sth getbypolicy MSSQL <POLICY name>
   ```

4 Enable or disable the stream handler for a policy and a client.

```
cacontrol --sth update MSSQL <Client name> <POLICY name>
enabled/disabled
```

5 Delete the stream handler settings for a policy and a client (returns to the default behavior). Microsoft SQL stream handler is enabled by default.

```
cacontrol --sth delete MSSQL <Client name> <POLICY name>
```

6 Query the settings for the stream handler for a policy and a client.

```
cacontrol --sth get MSSQL <Client name><POLICY name>
```

7 Verify that the Microsoft SQL stream handler is enabled or disabled.

On the NetBackup web UI, under the **Job Details** tab, **MSSQL Stream Handler enabled** is displayed.

Things to consider:

- When Client Direct setting is enabled, the Microsoft SQL stream handler is not used even if the **MS-SQL-Server** policy type is used. The **pdplugin** does not know the policy type when it runs on the client side.
  To enable the stream handler, use the `cacontrol` command to enable the Microsoft SQL stream handler for the policy or the client and policy.
- Do not enable the NetBackup compression settings when the storage type is MSDP. This setting causes deduplication loss even if the Microsoft SQL stream handler is enabled or disabled.
- The Microsoft SQL compression setting is provided by Microsoft SQL Server. It compresses the SQL data. When Microsoft SQL native compression is enabled, the deduplication rate may drop.
- Microsoft SQL stream handler works well with the Microsoft SQL Server Transparent Data Encryption (TDE). Microsoft SQL TDE only encrypts the SQL data and the SQL page structure does not change. The deduplication is not lost when the Microsoft SQL Server TDE is enabled.

MSDP deployment best practices

Because Cohesity recommends minimum host and network requirements only, deduplication performance may vary greatly depending on your environment. Cohesity provides best-practice guidelines to help you use deduplication effectively regardless of the capabilities of your hosts.

Cohesity recommends that you consider the following practices when you implement NetBackup deduplication.

Use fully qualified domain names

Cohesity recommends that you use fully qualified domain names for your NetBackup servers (and by extension, your deduplication servers). Fully qualified domain names can help to avoid host name resolution problems, especially if you use client-side deduplication.

Deduplication servers include the storage server and the load balancing servers (if any).

See "MSDP media write error (84)" on page 784.

About scaling MSDP

You can scale deduplication processing to improve performance by using load balancing servers or client deduplication or both.

If you configure load balancing servers, those servers also perform deduplication. The deduplication storage server still functions as both a deduplication server and as a storage server. NetBackup uses standard load balancing criteria to select a load balancing server for each job. However, deduplication fingerprint calculations are not part of the load balancing criteria.

To completely remove the deduplication storage server from deduplication duties, do the following for every storage unit that uses the deduplication disk pool:

- Select **Only use the following media servers.**
- Select all of the load balancing servers but do not select the deduplication storage server.

The deduplication storage server performs storage server tasks only: storing and managing the deduplicated data, file deletion, and optimized duplication.

If you configure client deduplication, the clients deduplicate their own data. Some of the deduplication load is removed from the deduplication storage server and loading balancing servers.

Cohesity recommends the following strategies to scale MSDP:

- For the initial full backups of your clients, use the deduplication storage server. For subsequent backups, use load balancing servers.
- Enable client-side deduplication gradually.
  If a client cannot tolerate the deduplication processing workload, be prepared to move the deduplication processing back to a server.

Send initial full backups to the storage server

If you intend to use load balancing servers or client deduplication, use the storage server for the initial full backups of the clients. Then, send subsequent backups through the load balancing servers or use client deduplication for the backups. Doing so provides information about the total deduplication load. You can then allocate jobs to best balance the load among your hosts.

Deduplication uses the same fingerprint list regardless of which host performs the deduplication. So you can deduplicate data on the storage server first, and then subsequent backups by another host use the same fingerprint list. If the deduplication plug-in can identify the last full backup for the client and the policy combination, it retrieves the fingerprint list from the server. The list is placed in the fingerprint cache for the new backup.

See "About MSDP fingerprinting" on page 73.

Cohesity also recommends that you implement load balancing servers and client deduplication gradually. Therefore, it may be beneficial to use the storage server for backups while you implement deduplication on other hosts.

Increase the number of MSDP jobs gradually

Cohesity recommends that you increase the **Maximum concurrent jobs** value gradually. (The **Maximum concurrent jobs** is a storage unit setting.) Doing so provides information about the total deduplication load. The initial backup jobs (also known as initial seeding) require more CPU and memory than successive jobs. After initial seeding, the storage server can process more jobs concurrently. You can then gradually increase the jobs value over time.

Introduce MSDP load balancing servers gradually

Cohesity recommends that you add load balancing servers only after the storage server reaches maximum CPU utilization. Then, introduce load balancing servers one at a time. It may be easier to evaluate how your environment handles traffic and easier to troubleshoot any problems with fewer hosts added for deduplication.

See "About MSDP storage servers" on page 35.

Because of the various factors, Cohesity recommends that you maintain realistic expectations about using multiple servers for deduplication. If you add one media server as a load balancing server, overall throughput should be faster. However, adding one load balancing server may not double the overall throughput rate, adding two load balancing servers may not triple the throughput rate, and so on.

If all of the following apply to your MSDP environment, your environment may be a good candidate for load balancing servers:

- The deduplication storage server is CPU limited on any core.
- Memory resources are available on the storage server.
- Network bandwidth is available on the storage server.
- Back-end I/O bandwidth to the deduplication pool is available.
- Other NetBackup media servers have CPU available for deduplication.

Gigabit Ethernet should provide sufficient performance in many environments. If your performance objective is the fastest throughput possible with load balancing servers, you should consider 10 Gigabit Ethernet.

See "Planning your MSDP deployment" on page 25.

See "MSDP deployment best practices" on page 50.

Implement MSDP client deduplication gradually

If you configure clients to deduplicate their own data, do not enable all of those clients at the same time. Implement client deduplication gradually, as follows:

- Use the storage server for the initial backup of the clients.
- Enable deduplication on only a few clients at a time.
  Doing so provides information about deduplication affects the clients other jobs. It also may be easier to evaluate how your environment handles traffic and easier to troubleshoot any problems

If a client cannot tolerate the deduplication processing workload, be prepared to move the deduplication processing back to the storage server.

Use MSDP compression and encryption

Do not use compression or encryption in a NetBackup policy; rather, use the compression or the encryption that is part of the deduplication process.

See "About MSDP compression" on page 108.

See "About MSDP encryption" on page 109.

About the optimal number of backup streams for MSDP

A backup stream appears as a separate job in the NetBackup Activity Monitor. Various methods exist to produce streams. In NetBackup, you can use backup policy settings to configure multiple streams. The NetBackup for Oracle agent lets you configure multiple streams; also for Oracle the RMAN utilities can provide multiple backup channels.

For client deduplication, the optimal number of backup streams is two.

Media server deduplication can process multiple streams on multiple cores simultaneously. For large datasets in applications such as Oracle, media server deduplication leverages multiple cores and multiple streams. Therefore, media server deduplication may be a better solution when the application can provide multiple streams or channels.

About storage unit groups for MSDP

You can use a storage unit group as a backup destination for NetBackup MSDP. All of the storage units in the group must have a **Media Server Deduplication Pool** as the storage destination.

Storage unit groups avoid a single point of failure that can interrupt backup service.

The best storage savings occur when a backup policy stores its data in the same deduplication destination disk pool instead of across multiple disk pools. For this reason, the **Failover** method for the **Storage unit selection** uses the least amount of storage. All of the other methods are designed to use different storage every time the backup runs. Cohesity recommends that you select the **Failover** method for the **Storage unit selection** type.

Table 2-10 MSDP requirements and limitations for storage unit groups

What	Description
Requirements	A group must contain storage units of one storage destination type only. That is, a group cannot contain both **Media Server Deduplication Pool** storage units and storage units with other storage types.
Limitations	NetBackup does not support the following for storage unit groups: ■ Optimized duplication of deduplicated data. If you use a storage unit group as a destination for optimized duplication of deduplicated data, NetBackup uses regular duplication. See “About MSDP optimized duplication within the same domain” on page 122. ■ Optimized synthetic backups. If NetBackup cannot produce the optimized synthetic backup, NetBackup creates the more data-movement intensive synthetic backup. See “About MSDP optimized synthetic backups” on page 44.

About protecting the MSDP data

Cohesity recommends the following methods to protect the deduplicated backup data:

- Use NetBackup optimized duplication to copy the images to another deduplication node at an off-site location.

  Optimized duplication copies the primary backup data to another deduplication pool. It provides the easiest, most efficient method to copy data off-site yet remain in the same NetBackup domain. You then can recover from a disaster that destroys the storage on which the primary copies reside by retrieving images from the other deduplication pool.

  See "Configuring MSDP optimized duplication within the same NetBackup domain" on page 129.

- Use NetBackup replication to copy the deduplicated data to another NetBackup domain off-site.

  See "Configuring MSDP replication to a different NetBackup domain" on page 137.

Cohesity also recommends that you back up the MSDP catalog.

See "About protecting the MSDP catalog" on page 201.

Save the MSDP storage server configuration

Cohesity recommends that you save the storage server configuration. Getting and saving the configuration can help you with recovery of your environment. For disaster recovery, you may need to set the storage server configuration by using a saved configuration file.

If you save the storage server configuration, you must edit it so that it includes only the information that is required for recovery.

See "About saving the MSDP storage server configuration" on page 195.

See "Saving the MSDP storage server configuration" on page 196.

See "Editing an MSDP storage server configuration file" on page 197.

Plan for disk write caching

Storage components may use hardware caches to improve read and write performance. Among the storage components that may use a cache are disk arrays, RAID controllers, or the hard disk drives themselves.

If your storage components use caches for disk write operations, ensure that the caches are protected from power fluctuations or power failure. If you do not protect against power fluctuations or failure, data corruption or data loss may occur.

Protection can include the following:

- A battery backup unit that supplies power to the cache memory so write operations can continue if power is restored within sufficient time.

- An uninterruptible power supply that allows the components to complete their write operations.

If your devices that have caches are not protected, Cohesity recommends that you disable the hardware caches. Read and write performance may decline, but you help to avoid data loss.

Chapter 3

Provisioning the storage

This chapter includes the following topics:

- About provisioning the storage for MSDP
- Do not modify MSDP storage directories and files
- About volume management for NetBackup MSDP

About provisioning the storage for MSDP

NetBackup requires that the storage is exposed as a directory path.

Provision the storage as follows:

Up to 64 TBs

400 TBs

How many storage instances you provision depends on your storage requirements for your backups. If your requirements are greater than one deduplication node can accommodate, you can configure more than one node.

See “About MSDP deduplication nodes” on page 27.

Optimized duplication and replication can also affect the number of nodes you provision.

See “About MSDP optimized duplication and replication” on page 45.

Other NetBackup requirements may affect how you provision the storage.

See “About MSDP storage and connectivity requirements” on page 30.

How to provision the storage is beyond the scope of the NetBackup documentation. Consult the storage vendor’s documentation.

See “About the NetBackup deduplication destination” on page 27.

See “Planning your MSDP deployment” on page 25.

Up to 64 TBs of storage

Provision the backup storage so that it appears as a single mount point to the operating system.

Because the storage requires a directory path, do not use only the root node (/) or drive letter (E:\) as the storage path. (That is, do not mount the storage as a root node (/) or a drive letter (E:\).

If you use a separate disk volume for the deduplication database, provision a 1-TB volume on a different mount point than the backup data storage.

400 TBs of storage

NetBackup supports 400 TBs of storage in a single **Media Server Deduplication Pool** on certain operating systems.

See “About MSDP capacity support and hardware requirements” on page 28.

Before you configure the MSDP storage server, you must provision the volumes. Each volume must conform to the following items:

- Formatted with a file system that NetBackup supports for MSDP. The same file system must be used for all volumes.
- Reside on a separate disk from the other volumes that you allocate for the MSDP storage.
- Mounted on a separate mount point on the computer that you want to use as the MSDP storage server.
  Cohesity recommends that you use a descriptive naming convention for the mount point names.

Steps to configure the 400 TB MSDP using 50-TB volumes

1. Create, format, and mount nine new file systems. One file system must have 1-TB storage space and the other eight file systems must have 50-TB storage space each.
2. Mount the 1-TB file system at `/msdp/cat` and the 50-TB file systems on `/msdp/vol0`, `/msdp/vol1` and so on until each volume is mounted.
3. Create a touch file `/etc/nbapp-release` if it does not exist.
4. Create a subdirectory named **data** under each mounted volume. For example, `/msdp/vol0/data`, `/msdp/vol1/data`, `/msdp/vol2/data`, and so on.

5 Configure an MSDP storage server. Ensure that the **Use alternate path for deduplication database** option is selected. Provide the storage path as `/msdp/vol0/data` and the database path as `/msdp/cat`.

6 Add additional 50-TB file systems to the deduplication pool:

```
/usr/openv/pdde/pdcr/bin/crcontrol --dsaddpartition /msdp/vol1/data
/usr/openv/pdde/pdcr/bin/crcontrol --dsaddpartition /msdp/vol2/data
till volume 07...
/usr/openv/pdde/pdcr/bin/crcontrol --dsaddpartition /msdp/vol7/data
```

If a non-root user is used for the NetBackup media server, run the following command to change the owner of the newly created volumes to the NetBackup media server service user:

```
chown -R <NBU-service-user>:root <MSDP-volume-path>
```

7 Review the following command output to verify the created volumes:

```
/usr/openv/pdde/pdcr/bin/crcontrol --dsstat 2 | grep Mount
Mount point count: 7
```

See "Resizing the MSDP storage partition" on page 564.

Do not modify MSDP storage directories and files

Unless you are directed to do so by the NetBackup documentation or by a Cohesity support representative, do not do the following:

- Add files to the deduplication storage directories or database directories.
- Delete files from the deduplication storage directories or database directories.
- Modify files in the deduplication storage directories or database directories.
- Move files within the deduplication storage directories or database directories.
- Change the permissions of the directories and files within the deduplication storage directories or database directories.

Failure to follow these directives can result in operational failures and data loss.

About volume management for NetBackup MSDP

If you use a tool to manage the volumes for NetBackup **Media Server Deduplication Pool** storage, Cohesity recommends that you use the Arctera InfoScale Storage. InfoScale Storage includes the Arctera Volume Manager and the Arctera File System.

For supported systems, see the InfoScale hardware compatibility list at the Arctera website:

http://www.veritas.com/

Note: Although InfoScale Storage supports NFS, NetBackup does not support NFS targets for **Media Server Deduplication Pool** storage. Therefore, **Media Server Deduplication Pool** does not support NFS with InfoScale Storage.

Chapter 4

Configuring deduplication

This chapter includes the following topics:

- Configuring media server deduplication in NetBackup
- Configuring MSDP client-side deduplication
- About the MSDP Deduplication Multi-Threaded Agent
- About MSDP fingerprinting
- Enabling 400 TB support for MSDP
- Configuring a storage server for a Media Server Deduplication Pool
- About disk pools for NetBackup deduplication
- Configuring a Media Server Deduplication Pool storage unit
- Configuring client attributes for MSDP client-side deduplication
- About MSDP compression
- About MSDP encryption
- About MSDP Encryption using NetBackup Key Management Server service
- About MSDP Encryption using external KMS server
- Configuring optimized synthetic backups for MSDP
- About a separate network path for MSDP duplication and replication
- About MSDP optimized duplication within the same domain
- About MSDP replication to a different domain
- Configuring MSDP replication to a different NetBackup domain

- About configuring MSDP optimized duplication and replication bandwidth
- About performance tuning of optimized duplication and replication for large images
- About performance tuning of optimized duplication and replication for MSDP cloud
- About storage lifecycle policies
- About MSDP backup policy configuration
- Resilient network properties
- Adding an MSDP load balancing server
- About variable-length deduplication on NetBackup clients
- About the MSDP pd.conf configuration file
- About the MSDP contentrouter.cfg file
- About saving the MSDP storage server configuration
- Setting the MSDP storage server configuration
- About the MSDP host configuration file
- Deleting an MSDP host configuration file
- Resetting the MSDP registry
- About protecting the MSDP catalog
- About MSDP FIPS compliance
- Configuring the NetBackup client-side deduplication to support multiple interfaces of MSDP
- About MSDP multi-domain support
- About MSDP application user support
- About MSDP mutli-domain VLAN Support
- About NetBackup WORM storage support for immutable and indelible data
- Running MSDP services with the non-root user
- Running MSDP commands with the non-root user

Configuring media server deduplication in NetBackup

Table 4-1 describes the configuration tasks.

The NetBackup Administrator's Guide, Volume I describes how to configure a base NetBackup environment.

Table 4-1 MSDP configuration tasks

Task	Procedure
Configure a deduplication storage server	How many storage servers you configure depends on: your storage requirements and on whether or not you use optimized duplication or replication. When you configure a storage server, the UI also lets you configure a disk pool and a storage unit. See "About MSDP storage servers" on page 35. See "MSDP storage path properties" on page 95. See "About MSDP optimized duplication and replication" on page 45. Which type of storage server to configure depends on the storage destination. See "About the NetBackup deduplication destination" on page 27. See "Configuring a storage server for a Media Server Deduplication Pool" on page 93.
Configure a disk pool	If you already configured a disk pool when you configured the storage server, you can skip this step. How many disk pools you configure depends on: your storage requirements and on whether or not you use optimized duplication or replication. See "About disk pools for NetBackup deduplication" on page 97. See "Configuring a disk pool for deduplication" on page 98.
Configure a storage unit	See "Configuring a Media Server Deduplication Pool storage unit" on page 102.
Configure a backup policy	Use the deduplication storage unit as the destination for the backup policy. If you configured replication, use the storage lifecycle policy as the storage destination. See "About MSDP backup policy configuration" on page 169.
Configure optimized duplication copy	Optimized duplication is optional. See "About MSDP optimized duplication within the same domain" on page 122.
Configure replication	Replication is optional. See "About MSDP replication to a different domain" on page 136.

Table 4-1 MSDP configuration tasks *(continued)*

Task	Procedure
Configure MSDP client-side deduplication	See "Configuring MSDP client-side deduplication" on page 65.
Create NetBackup log file directories on the primary server and the media servers	See "NetBackup MSDP log files" on page 767. See "Creating NetBackup log file directories for MSDP" on page 766.
Configure the Deduplication Multi-Threaded Agent behavior	The Deduplication Multi-Threaded Agent uses the default configuration values that control its behavior. You can change those values if you want to do so. See "About the MSDP Deduplication Multi-Threaded Agent" on page 66. See "Configuring the Deduplication Multi-Threaded Agent behavior" on page 67. See "Configuring deduplication plug-in interaction with the Multi-Threaded Agent" on page 72.
Configure the fingerprint cache behavior	Configuring the fingerprint cache behavior is optional. See "About the MSDP fingerprint cache" on page 73. See "Configuring the MSDP fingerprint cache behavior" on page 74.
Enable support for 400 TB MSDP	Before you configure a storage server that hosts a 400 TB **Media Server Deduplication Pool**, you must enable support for that size storage. See "Enabling 400 TB support for MSDP" on page 89.
Create the data directories for 400 TB support	For a 400 TB **Media Server Deduplication Pool**, you must create the data directories under the mount points for the storage directories. See "Creating the data directories for 400 TB MSDP support" on page 89.
Add the other volumes for 400 TB support	For a 400 TB **Media Server Deduplication Pool**, you must add the second and third volumes to the disk pool. See "Adding volumes to a 400 TB Media Server Deduplication Pool" on page 90.
Enable encryption	Encryption is optional. See "Configuring encryption for MSDP local storage volume" on page 109.
Configure optimized synthetic backups	Optimized synthetic backups are optional. See "Configuring optimized synthetic backups for MSDP" on page 120.
Configure MSDP restore behavior	Optionally, you can configure NetBackup to bypass media servers during restores. See "How MSDP restores work" on page 565. See " Configuring MSDP restores directly to a client" on page 565.

Table 4-1 MSDP configuration tasks *(continued)*

Task	Procedure
Specify advanced deduplication settings	Advanced settings are optional. See "About the MSDP pd.conf configuration file" on page 178. See "Editing the MSDP pd.conf file" on page 179. See "MSDP pd.conf file parameters" on page 179.
Protect the MSDP data and catalog	See "About protecting the MSDP data" on page 54. See "About protecting the MSDP catalog" on page 201.

Configuring MSDP client-side deduplication

This topic describes how to configure client deduplication in NetBackup. Media server deduplication must be configured before you can configure client-side deduplication.

See "Configuring media server deduplication in NetBackup" on page 63.

Table 4-2 Client deduplication configuration tasks

Task	Procedure
Configure media server deduplication	See "Configuring media server deduplication in NetBackup" on page 63.
Learn about client deduplication	See "About NetBackup Client Direct deduplication" on page 38.
Configure a resilient connection for remote office clients	Resilient connections are optional. See "About MSDP remote office client deduplication" on page 40. See "Resilient network properties" on page 170. See "Specify resilient connections for clients" on page 172.
Enable client-side deduplication	See "Configuring client attributes for MSDP client-side deduplication" on page 106.
Configure remote client fingerprint cache seeding	Configuring remote client fingerprint cache seeding is optional. See "Configuring MSDP fingerprint cache seeding on the client" on page 83. See "About seeding the MSDP fingerprint cache for remote client deduplication" on page 81. See "Configuring MSDP fingerprint cache seeding on the storage server" on page 84.

Table 4-2 Client deduplication configuration tasks *(continued)*

Task	Procedure
Configure client-direct restores	Configuring client-direct restores is optional. If you do not do so, restores travel through the NetBackup media server components. See " Configuring MSDP restores directly to a client" on page 565.

About the MSDP Deduplication Multi-Threaded Agent

The MSDP deduplication process can use a Multi-Threaded Agent for most data sources. The Multi-Threaded Agent runs alongside the deduplication plug-in on both the clients and the media servers. The agent uses multiple threads for asynchronous network I/O and CPU core calculations. During a backup, this agent receives data from the deduplication plug-in through shared memory and processes it using multiple threads to improve throughput performance. When inactive, the agent uses minimal resources.

The NetBackup Deduplication Multi-Threaded Agent improves backup performance for any host that deduplicates data: the storage server, load balancing servers, or clients that deduplicate their own data. For each host on which you want to use the Multi-Threaded Agent, you must configure the deduplication plug-in to use it.

The Deduplication Multi-Threaded Agent uses the default configuration values that control its behavior. You can change those values if you want to do so. The following table describes the Multi-Threaded Agent interactions and behaviors. It also provides links to the topics that describe how to configure those interactions and behaviors.

Table 4-3 Interactions and behaviors

Interaction	Procedure
Multi-Threaded Agent behavior and resource usage	See "Configuring the Deduplication Multi-Threaded Agent behavior" on page 67.
Whether or not the deduplication plug-in sends backups to the Multi-Threaded Agent	See "Configuring deduplication plug-in interaction with the Multi-Threaded Agent" on page 72.
The clients that should use the Deduplication Multi-Threaded Agent for backups	See "Configuring deduplication plug-in interaction with the Multi-Threaded Agent" on page 72.

Table 4-3 Interactions and behaviors *(continued)*

Interaction	Procedure
The backup policies that should use the Deduplication Multi-Threaded Agent	See "Configuring deduplication plug-in interaction with the Multi-Threaded Agent" on page 72.

Table 4-4 describes the operational notes for MSDP multithreading. If the Multi-Threaded Agent is not used, NetBackup uses the single-threaded mode.

Table 4-4 Multi-Threaded Agent requirements and limitations

Item	Description
Supported systems	NetBackup supports the Multi-Threaded Agent on Linux, Solaris, AIX, and Windows operating systems.
Unsupported use cases	NetBackup does not use the Multi-Threading Agent for the following use cases: ■ Virtual synthetic backups ■ `SEGKSIZE` is greater than 128 (`pd.conf` file) ■ `DONT_SEGMENT_TYPES` enabled (`pd.conf` file) ■ `MATCH_PDRO = 1` (`pd.conf` file) See "MSDP pd.conf file parameters" on page 179.
Policy-based compression or encryption	If NetBackup policy-based compression or encryption is enabled on the backup policy, NetBackup does not use the Deduplication Multi-Threaded Agent. Cohesity recommends that you use the MSDP compression and encryption rather than NetBackup policy-based compression and encryption. See "About MSDP compression" on page 108. See "About MSDP encryption" on page 109.

Configuring the Deduplication Multi-Threaded Agent behavior

The `mtstrm.conf` configuration file controls the behavior of the NetBackup Deduplication Multi-Threaded Agent.

See "About the MSDP Deduplication Multi-Threaded Agent" on page 66.

If you change the `mtstrm.conf` file on a host, it changes the settings for that host only. If you want the same settings for all of the hosts that deduplicate data, you must change the `mtstrm.conf` file on all of the hosts.

To configure the Multi-Threaded Agent behavior

1. Use a text editor to open the `mtstrm.conf` file.

   The `mtstrm.conf` file resides in the following directories:

   - UNIX: `/usr/openv/lib/ost-plugins/`
   - Windows: *`install_path`*`\Veritas\NetBackup\bin\ost-plugins`

2. To change a behavior, specify a new value.

   See "MSDP mtstrm.conf file parameters" on page 68.

3. Save and close the file.

4. Restart the Multi-Threaded Agent on the host, as follows:

   - On UNIX:

     ```
     /usr/openv/pdde/pdag/bin/mtstrmd -terminate
     /usr/openv/pdde/pdag/bin/mtstrmd
     ```

   - On Windows, use the Windows Services manager. The service name is NetBackup Deduplication Multi-Threaded Agent.

MSDP mtstrm.conf file parameters

The `mtstrm.conf` configuration file controls the behavior of the Deduplication Multi-threaded Agent. The default values balance performance with resource usage.

A procedure exists that describes how to configure these parameters.

The `pd.conf` file resides in the following directories:

- (UNIX) `/usr/openv/lib/ost-plugins/`
- (Windows) *`install_path`*`\Veritas\NetBackup\bin\ost-plugins`

See "Configuring the Deduplication Multi-Threaded Agent behavior" on page 67.

The `mtstrm.conf` file is comprised of three sections. The parameters must remain within their sections. For descriptions of the parameters, see the following sections:

- Logging parameters
- Process parameters
- Threads parameters

The `mtstrm.conf` file resides in the following directories:

- UNIX: `/usr/openv/lib/ost-plugins/`
- Windows: *`install_path`*`\Veritas\NetBackup\bin\ost-plugins`

Logging parameters

The following table describes the logging parameters of the `mtstrm.conf` configuration file.

Table 4-5 Logging parameters (mtstrm.conf file)

Logging Parameter	Description
`LogPath`	The directory in which the `mtstrmd.log` files are created. Default values: ■ Windows: `LogPath=install_path\Veritas\pdde\\..\netbackup\logs\pdde` ■ UNIX: `LogPath=/var/log/puredisk`
`Logging`	Specify what to log: Default value: `Logging=short,thread`. Possible values: `minimal: Critical, Error, Authentication, Bug` `short : all of the above plus Warning` `long : all of the above plus Info` `verbose: all of the above plus Notice` `full : all of the above plus Trace messages (everything)` `none : disable logging` To enable or disable other logging information, append one of the following to the logging value, without using spaces: `,thread : enable thread ID logging.` `,date : enable date logging.` `,timing : enable high-resolution timestamps` `,silent : disable logging to console`
`Retention`	How long to retain log files (in days) before NetBackup deletes them. Default value: `Retention=7`. Possible values: 0-9, inclusive. Use 0 to keep logs forever.
`LogMaxSize`	The maximum log size (MB) before NetBackup creates a new log file. The existing log files that are rolled over are renamed `mtstrmd.log.<date/time stamp>` Default value: `LogMaxSize=500`. Possible value: 1 to the maximum operating system file size in MBs, inclusive.

Process parameters

The following table describes the process parameters of the `mtstrm.conf` configuration file.

Table 4-6 Process parameters (mtstrm.conf file)

Process Parameter	Description
`MaxConcurrentSessions`	The maximum number of concurrent sessions that the Multi-Threaded Agent processes. If it receives a backup job when the `MaxConcurrentSessions` value is reached, the job runs as a single-threaded job. By default, the deduplication plug-in sends backup jobs to the Multi-Threaded Agent on a first-in, first-out basis. However, you can configure which clients and which backup policies the deduplication plug-in sends to the Multi-Threaded Agent. The `MTSTRM_BACKUP_CLIENTS` and `MTSTRM_BACKUP_POLICIES` parameters in the `pd.conf` control the behavior. Filtering the backup jobs that are sent to the Multi-Threaded Agent can be very helpful on the systems that have many concurrent backup jobs. See "MSDP pd.conf file parameters" on page 179. Default value: `MaxConcurrentSessions=` (calculated by NetBackup; see the following paragraph). NetBackup configures the value for this parameter during installation or upgrade. The value is the hardware concurrency value of the host divided by the `BackupFpThreads` value (see Table 4-7). (For the purposes of this parameter, the *hardware concurrency* is the number of CPUs or cores or hyperthreading units.) On media servers, NetBackup may not use all hardware concurrency for deduplication. Some may be reserved for other server processes. For more information about hardware concurrency, see the `pd.conf` file `MTSTRM_BACKUP_ENABLED` parameter description. See "MSDP pd.conf file parameters" on page 179. Possible values: 1-32, inclusive. **Warning:** Cohesity recommends that you change this value only after careful consideration of how the change affects your system resources. With default configuration values, each session uses approximately 120 to 150 MBs of memory. The memory that is used is equal to (`BackupReadBufferCount` * `BackupReadBufferSize`) + (3 * `BackupShmBufferSize`) + `FpCacheMaxMbSize` (if enabled).

Table 4-6 Process parameters (mtstrm.conf file) *(continued)*

Process Parameter	Description
`BackupShmBufferSize`	The size of the buffers (MB) for shared memory copying. This setting affects three buffers: The shared memory buffer itself, the shared memory receive buffer in the `mtstrmd` process, and the shared memory send buffer on the client process. Default value: `BackupShmBufferSize=2` (UNIX) or `BackupShmBufferSize=8` (Windows). Possible values: 1-16, inclusive.
`BackupReadBufferSize`	The size (MB) of the memory buffer to use per session for read operations from a client during a backup. Default value: `BackupReadBufferSize=32` . Possible values: 16-128, inclusive.
`BackupReadBufferCount`	The number of memory buffers to use per session for read operations from a client during a backup. Default value: `BackupReadBufferCount=3`. Possible values: `1` to `10`, inclusive.
`BackupBatchSendEnabled`	Determines whether to use batch message protocols to send data to the storage server for a backup. Default value: `BackupBatchSendEnabled=1`. Possible values: `0` (disabled) or `1` (enabled).
`FpCacheMaxMbSize`	The maximum amount of memory (MB) to use per session for fingerprint caching. Default value: `FpCacheMaxMbSize=1024`. Possible values: 0-1024, inclusive.
`SessionCloseTimeout`	The amount of time to wait in seconds for threads to finish processing when a session is closed before the agent times-out with an error. Default value: 180. Possible values: 1-3600.
`SessionInactiveThreshold`	The number of minutes for a session to be idle before NetBackup considers it inactive. NetBackup examines the sessions and closes inactive ones during maintenance operations. Default value: 480. Possible values: 1-1440, inclusive.

Threads parameters

The following table describes the threads parameters of the `mtstrm.conf` configuration file.

Table 4-7 Threads parameters (mtstrm.conf file)

Threads Parameter	Description
`BackupFpThreads`	The number of threads to use per session to fingerprint incoming data. Default value: `BackupFpThreads=` (calculated by NetBackup; see the following explanation). NetBackup configures the value for this parameter during installation or upgrade. The value is equal to the following hardware concurrency threshold values. ■ Windows and Linux: The threshold value is 2. ■ Solaris: The threshold value is 4. For more information about hardware concurrency, see the `pd.conf` file `MTSTRM_BACKUP_ENABLED` parameter description. See "MSDP pd.conf file parameters" on page 179.
`BackupSendThreads`	The number of threads to use per session to send data to the storage server during a backup operation. Default value: `BackupSendThreads=1` for servers and `BackupSendThreads=2` for clients. Possible values: 1-32, inclusive.
`MaintenanceThreadPeriod`	The frequency at which NetBackup performs maintenance operations, in minutes. Default value: 720. Possible values: 0-10080, inclusive. Zero (0) disables maintenance operations.

Configuring deduplication plug-in interaction with the Multi-Threaded Agent

You can control the interaction between the NetBackup deduplication plug-in and the Multi-Threaded Agent. Several settings in the `pd.conf` file on a host control the interaction. A change in a `pd.conf` file changes the settings for that host only. If you want the same settings for all of the hosts that deduplicate data, you must change the `pd.conf` file on all of the hosts.

See "About the MSDP pd.conf configuration file" on page 178.

To configure deduplication plug-in interaction with the Multi-Threaded Agent

1 Use a text editor to open the `pd.conf` file.

The pd.conf file resides in the following directories:

- (UNIX) /usr/openv/lib/ost-plugins/
- (Windows) *install_path*\Veritas\NetBackup\bin\ost-plugins

2 To change a setting, specify a new value. The following are the settings that control the interaction:

- MTSTRM_BACKUP_CLIENTS
- MTSTRM_BACKUP_ENABLED
- MTSTRM_BACKUP_POLICIES
- MTSTRM_IPC_TIMEOUT

These settings are described in another topic.

See "MSDP pd.conf file parameters" on page 179.

3 Save and close the file.

4 Restart the NetBackup Remote Manager and Monitor Service (nbrmms) on the host.

About MSDP fingerprinting

NetBackup uses a unique identifier to identify each file and each file segment that is backed up. The deduplication plug-in reads the backup image and separates the image into files. The plug-in separates the files into segments. For each segment, the plug-in calculates the hash key (or *fingerprint*) that identifies each data segment. To create a hash, every byte of data in the segment is read and added to the hash.

To ensure that data is encrypted and secured with the highest standards, NetBackup uses the AES encryption algorithm and SHA-2 fingerprinting algorithm beginning with the 8.1 release. Specifically, MSDP uses AES-256 and SHA-512/256.

About the MSDP fingerprint cache

NetBackup uses *fingerprints* to identify the data segments in the backup data. NetBackup writes only unique data segments to a **Media Server Deduplication Pool**. If a segment is already in storage meeting the preconfigured data locality requirements, NetBackup does not store it again.

See "About MSDP fingerprinting" on page 73.

The storage server maintains an index cache of the fingerprints in RAM. After startup, the NetBackup Deduplication Engine (spoold) loads fingerprints into this cache in the background. The background loading continues until all required

fingerprints are loaded or the configured cache capacity is reached. The fingerprints in the cache are then updated as jobs run.

For each backup job, the MSDP OST (OpenStorage) plug-in requests the fingerprints of the last backup of the current client from the MSDP server. The fingerprints are stored in a size-limited cache in the memory of the MSDP OST plug-in. It's called the client-side fingerprint cache. The plug-in checks this cache for a fingerprint first. After a missed lookup, it requests the fingerprint lookup with the MSDP server.

You can configure the cache loading behavior.

See "Configuring the MSDP fingerprint cache behavior" on page 74.

You can also control the fingerprint cache seeding for clients.

See "About seeding the MSDP fingerprint cache for remote client deduplication" on page 81.

Configuring the MSDP fingerprint cache behavior

You can configure the cache loading behavior.

See "About the MSDP fingerprint cache" on page 73.

See "About the MSDP contentrouter.cfg file" on page 194.

To configure MSDP fingerprint cache behavior

1. On the storage server, open the `contentrouter.cfg` file in a text editor; it resides in the following directory:
   - (UNIX) *`storage_path`*`/etc/puredisk`
   - (Windows) *`storage_path`*`\etc\puredisk`
2. Edit the parameters that control the behavior.

   See "MSDP fingerprint cache behavior options" on page 74.

MSDP fingerprint cache behavior options

The following tables describe the parameters that control MSDP fingerprint cache behavior, which are configured in the `contentrouter.cfg` file.

See "About the MSDP contentrouter.cfg file" on page 194.

Table 4-8 Cache size parameters

Behavior	Description
`MaxCacheSize`	The percentage of RAM to use for the fingerprint cache. This parameter is configured under the `Cache` section in the `contentrouter.cfg` file. **Note:** If P/S cache is enabled, use the `UsableMemoryLimit` parameter instead.
`UsableMemoryLimit`	For the systems that use P/S cache, the local volume and cloud volumes share the same S-cache and P-cache size, and the overall memory is limited by `UsableMemoryLimit` parameter under the `Cache` section in the `contentrouter.cfg` file. See "About sampling and predictive cache" on page 86.

Table 4-9 Cache load parameters

Parameter	Description
`FPCacheWithImageSet`	Controls the source from which the NetBackup Deduplication Engine (`spoold`) loads fingerprints into the fingerprint cache in the background after startup. If set to yes, `spoold` loads fingerprints from the most recent backups of existing clients. See "About the last backups of clients" on page 77. If set to no: ■ When `EnableLocalPredictiveSamplingCache=false`, fingerprints are loaded from data containers. ■ When `EnableLocalPredictiveSamplingCache=true`, only sampled fingerprints are loaded into memory at startup. No background fingerprint loading occurs after startup. The default value is **yes** when `EnableLocalPredictiveSamplingCache=true`; otherwise no.

Table 4-9 Cache load parameters *(continued)*

Parameter	Description
`LoadImageSetThreads`	Applicable when `FPCacheWithImageSet=yes`. Controls the maximum number of concurrent threads that are used to read fingerprints of the last images from one DataStore. The default value is 1.
`LoadImageSetTaskWaitTime`	Applicable when `FPCacheWithImageSet=yes`. Controls the maximum time (in seconds) that a job waits for its client's fingerprints to finish loading. This helps reduce the impact on deduplication efficiency when a job starts before fingerprint loading completes. If not specified, the default value is taken from `LastimageLoadTimeout` under the `Cache` section, which is 600 seconds. Set this value to 0 to disable waiting. Disabling waiting may negatively affect deduplication rate.

Use the `crcontrol` command with the `--fpcachestat` option to view fingerprint cache statistics, background loading progress, and related information.

- UNIX: `/usr/openv/pdde/pdcr/bin/crcontrol --fpcachestat [--json]`
- Windows: `C:\Program Files\Veritas\pdde\crcontrol --fpcachestat [--json]`

When `FPCacheWithImageSet` is enabled, background fingerprint loading messages are logged to `<storage>/log/spoold/LastImageFPLoader.log`. After the fingerprint loading process completes, the NetBackup Deduplication Engine records the completion status in `<storage>/log/spoold/spoold.log`.

One of the following messages is logged to indicate the result of the cache loading process:

- `Data Store nodes have completed cache loading (Complete Cache)`
- `Data Store nodes have completed cache loading (Incomplete Cache)`

About the last backups of clients

The NetBackup Deduplication Manager (spad) tracks the most recent backups of the clients. By default, these last backups are recorded in a cache file named __dofpcachefile__ for each client and policy combination.

The tracked last backups are used in the following scenarios:

- To populate the client-side fingerprint cache when a backup job for a client starts.
- To populate the server-side fingerprint cache when the FPCacheWithImageSet option is enabled in contentrouter.cfg

See "MSDP fingerprint cache behavior options" on page 74.

The following table describes the parameters that control the behavior of how the last backups are used. These parameters are defined under the StorageDatabase section in the <storage_path>/etc/puredisk/spa.cfg file.

Table 4-10 Last backup parameters

Parameter	Description
LastFullMaxImageCount	The maximum number of last images that return to a client in a job. The default value is 100.
LastFullMaxImageCountSmallDO	The maximum number of small fragments that return to a client in a job. The default value is 32.
LastFullSmallDOSize	The value defines a small fragment, in bytes. The default value is 20971520 (20 MiB).

About dynamic last images

In most environments, the tracked last backups work well without tuning. However, in environments with specialized workloads or clients, the static configuration may not suit all scenarios. In such cases, tuning the parameters alone may not be sufficient and can affect client-side deduplication efficiency.

Client-side deduplication efficiency is especially important in high-latency environments, such as NetBackup Client Direct backups, optimized duplication, or replication. In some cases, it can also affect the overall deduplication rate and performance for the client.

To address these situations, NetBackup 11.2 provides the `cacontrol` command to configure the rules that determine the selection of the last backups for a specific client. This capability enables on-demand, dynamic searches of the MSDP catalog instead of relying solely on tracked last backups. If no rule is configured, or if a configured rule does not return any results, the system falls back to using the tracked last backups.

Use the `cacontrol` command with the `--dynamicLastImages` option to configure and list the rules for the clients.

Table 4-11

Command	Description
`cacontrol --dynamicLastImages getRules [rulesFile]`	Returns the configured custom rules. If no rules are configured, the built-in template is returned. If `rulesFile` is specified, the output is written to that file. Otherwise, it is written to standard output.
`cacontrol --dynamicLastImages setRules rulesFile`	Applies the JSON rules that are defined in `rulesFile`.
`cacontrol --dynamicLastImages editRules`	(Linux only) Opens the current rules in the vim editor. Changes are applied when the file is saved and the editor is closed. The option combines the `getRules` and `setRules` options with the vim text editor. When the `editRules` option cannot be used (for example, on Windows), combine the commands with the `getRules` option, editing, and the `setRules` option.
`cacontrol --dynamicLastImages suggestRule dsid client policy`	Generates a suggested rule for a specified client based on existing MSDP catalog information. You can start with the suggestion to configure, monitor, and tune the rule for a client. **Note:** Suggested rules may not be optimal. Review and adjust parameters before use. Monitor backup results and tune the rules further as needed.

About Rule format

The rules are managed in a JSON format as with the following example.

```
{
    "defaults": {
        "full_image_search_depth": 1000,
        "affinity_time_gap_sec": 600,
        "affinity_size_gap_ratio": 0.1,
        "diversity_scale": 2,
        "max_imagecluster_candidates_no_source_size": 14,
        "max_imagecluster_candidates_with_source_size": 2,
        "max_incremental_images": 20
    },
    "source_list": [
        {
            "path": "SAMPLE_DSID/SAMPLE_CLIENT/SAMPLE_POLICY",
            "params": {}
        }
    ]
}
```

The configuration consists of two main sections `defaults` and `source_list`.

The `defaults` section defines the default values that apply to all rules unless overridden.

Table 4-12 The default parameters

Parameter	Default	Limits	Description
`full_image_search_depth`	1000	1–1000	Specifies how far back to search for full backup images.
`affinity_time_gap_sec`	600	60–43200	Maximum time gap (in seconds) for grouping backups into the same time cluster.
`affinity_size_gap_ratio`	0.1	0.01–10.0	Determines the acceptable size variation when source size is known, or diversity when unknown.
`diversity_scale`	2	0–30	Limits the number of distinct size clusters when source size is unknown.
`max_imagecluster_candidates_no_source_size`	14	1–30	Maximum candidate clusters when source size is unknown.

Table 4-12 The default parameters *(continued)*

Parameter	Default	Limits	Description
`max_imagecluster_candidates_with_source_size`	2	1–30	Maximum candidate clusters when source size is known.
`max_incremental_images`	20	0–100	Maximum number of the incremental images that are included after the selected full backup.

The `source_list` section defines the data sources to which dynamic last image rules apply. Each entry specifies a data source and, optionally, parameter overrides that differ from the default values.

The following table lists the fields that are included in each `source_list` entry.

Table 4-13 The `source_list` fields

Field	Description
`path`	Defines the data source to which the current rule applies, in the format DSID/Client/Policy. ■ Use exactly two slashes to separate the three segments. For example, 2/MyClient/MyPolicy, 3/*/MyPolicy. ■ You can use * in any segment to represent all values. For example, ***/ */MyPolicy**. Wildcards must be entire segments; partial patterns (such as **MyPolicy***) are not supported. ■ Entries with invalid paths are ignored. For example, missing a slash. Check the logs if a rule does not apply.
`Params` (optional)	Same parameter names as in `defaults`. If present, they override the defaults only for this path. Omit any parameter you do not want to override.

Note: If there are no deduplication efficiency issues, configuring dynamic rules is not required. Avoid creating a large number of rules, as excessive configuration may result in server-side request rejections.

When a dynamic last image rule is applied for a client, the following message is logged in the `spad` session log for the `action=getlastimage` request:

```
Got dynamic last images for DSID/Client/Policy
```

If the `spad` trace log is enabled, additional details are logged, including information about the selected full and incremental images.

About seeding the MSDP fingerprint cache for remote client deduplication

Cohesity provides a method for *seeding* the fingerprint cache for a new client. The use case that benefits the most from seeding is the first backup of a remote client over a high latency network such as a WAN. The performance of the first backup is then similar to the performance of an existing client.

An important consideration is the client from which to seed the cache. When you choose a similar client, consider the following:

- If most of the information is the operating system files, use any client with the same operating system.
- If most of the information is data, finding a client with the same data may be unlikely. Therefore, consider physically moving a copy of the data to the datacenter. Back up that data on a similar client, and then use that client and policy for the seed.
- The more similar the clients are, the greater the cache hit rate is.

Two methods exist to configure cache seeding. You can use either method. The following table describes the seeding configuration methods.

Table 4-14 Seeding configuration methods

Host on which to configure seeding	Description
On the client	Configure seeding on the client for one or only a few clients. See "Configuring MSDP fingerprint cache seeding on the client" on page 83.
On the storage server	The use case that benefits the most is many clients to seed, and they can use the fingerprint cache from a single host. See "Configuring MSDP fingerprint cache seeding on the storage server" on page 84.

To ensure that NetBackup uses the seeded backup images, the first backup of a client after you configure seeding must be a full backup with a single stream. Specifically, the following two conditions must be met in the backup policy:

- The **Attributes** tab **Allow multiple data streams** attribute must be unchecked.
- The backup selection cannot include any**NEW_STREAM** directives.

If these two conditions are not met, NetBackup may use multiple streams. If the **Attributes** tab **Limit jobs per policy** is set to a number less than the total number of streams, only those streams use the seeded images to populate the cache. Any streams that are greater than the **Limit jobs per policy** value do not benefit from seeding, and their cache hit rates may be close to 0%.

After the first backup, you can restore the original backup policy parameter settings.

The following items are example of informational messages that show that seeding occurred:

Activity Monitor Job Details

```
1/2/2015 2:18:23 AM - Info nbmaster1(pid=6340)
StorageServer=PureDisk:nbmaster1; Report=PDDO
Stats for (nbmaster1): scanned: 3762443 KB, CR
sent: 1022 KB, CR sent over FC: 0 KB, dedup:
100.0%, cache hits: 34364 (100.0%)
```

```
1/2/2015 2:18:24 AM - Info nbmaster1(pid=6340)
StorageServer=PureDisk:nbmaster1; Report=PDDO
Stats for (nbmaster1): scanned: 1 KB, CR sent:
0 KB, CR sent over FC: 0 KB, dedup: 100.0%
```

Deduplication plug-in log (`pdplugin.log`) on the client

```
01/02/15 02:15:17 [4452] [4884] [DEBUG] PDSTS:
cache_util_get_cache_dir: enter
db=/nbmaster1#1/2, scp='', bc=host1,
bp=seedfinal, bl=4096
```

```
01/02/15 02:15:17 [4452] [4884] [DEBUG] PDSTS:
cache_util_get_cache_dir: new backup, using
existing client seeding directory
```

```
01/02/15 02:15:17 [4452] [4884] [DEBUG] PDSTS:
cache_util_get_cache_dir: exit
db=/nbmaster1#1/2, scp='', bc=host1,
bp=seedfinal, bl=4096,
cachedir_buf='/nbmaster1#1/2/#pdseed/host1'
err=0
```

See "NetBackup MSDP log files" on page 767.

Deduplication proxy server log (`nbostpxy.log`) on the client	02:15:17.417 [4452.4884] [DEBUG] [dummy] [11:bptm:6340:nbmaster1] [DEBUG] PDSTS: cache_util_get_cache_dir: enter db=/nbmaster1#1/2, scp='', bc=host1, bp=seedfinal, bl=4096 02:15:17.433 [4452.4884] [DEBUG] [dummy] [11:bptm:6340:nbmaster1] [DEBUG] PDSTS: cache_util_load_fp_cache_nbu: enter dir_path=/nbmaster1#1/2/#pdseed/host1, t=16s, me=1024 02:15:17.449 [4452.4884] [DEBUG] [dummy] [11:bptm:6340:nbmaster1] [DEBUG] PDSTS: cache_util_load_fp_cache_nbu: adding 'nbmaster1_1420181254_C1_F1.img' to cache list (1) 02:15:17.449 [4452.4884] [DEBUG] [dummy] [11:bptm:6340:nbmaster1] [DEBUG] PDSTS: cache_util_load_fp_cache_nbu: opening /nbmaster1#1/2/#pdseed/host1/nbmaster1_1420181254_C1_F1.img for image cache (1/1) 02:15:29.585 [4452.4884] [DEBUG] [dummy] [11:bptm:6340:nbmaster1] [DEBUG] PDVFS: pdvfs_lib_log: soRead: segment c32b0756d491871c45c71f811fbd73af already present in cache. 02:15:29.601 [4452.4884] [DEBUG] [dummy] [11:bptm:6340:nbmaster1] [DEBUG] PDVFS: pdvfs_lib_log: soRead: segment 346596a699bd5f0ba5389d4335bc7429 already present in cache. See "NetBackup MSDP log files" on page 767.

See "About the MSDP fingerprint cache" on page 73.

Configuring MSDP fingerprint cache seeding on the client

Seeding on the client requires the following:

- A client name
- A policy name
- A date after which to stop using the similar client's fingerprint cache.

Information about when to use this seeding method and how to choose a client from which to seed is available.

See "About seeding the MSDP fingerprint cache for remote client deduplication" on page 81.

Warning: Do not use this procedure on the storage server or the load balancing server. If you do, it affects all clients that are backed up by that host.

To seed the MSDP fingerprint cache on the client

- Before the first backup of the remote client, edit the `FP_CACHE_CLIENT_POLICY` parameter in the `pd.conf` file on the remote client.

  Specify the setting in the following format:

  clienthostmachine,*backuppolicy*,*date*

clienthostmachine	The name of the existing similar client from which to seed the cache. **Note:** NetBackup treats long and short host names differently, so ensure that you use the client name as it appears in the policy that backs it up.
backuppolicy	The backup policy for that client.
date	The last date in mm/dd/yyyy format to use the fingerprint cache from the existing similar client. After this date, NetBackup uses the fingerprints from the client's own backup.

  See "Editing the MSDP pd.conf file" on page 179.

  See "MSDP pd.conf file parameters" on page 179.

Configuring MSDP fingerprint cache seeding on the storage server

On the storage server, the NetBackup `seedutil` utility creates a special seeding directory for a client. It populates the seeding directory with image references to another client and policy's backup images. The following is the pathname of the seeding directory:

database_path/databases/catalog/2/#pdseed/*client_name*

(By default, NetBackup uses the same path for the storage and the catalog; the *database_path* and the *storage_path* are the same. If you configure a separate path for the deduplication database, the paths are different.)

When a backup runs, NetBackup loads the fingerprints from the `#pdseed` directory for the client. (Assuming that no fingerprints exist for that client in the usual catalog location.)

Information about when to use this seeding method and how to choose a client from which to seed is available.

See "About seeding the MSDP fingerprint cache for remote client deduplication" on page 81.

To seed the fingerprint cache from the storage server

1. Before the first backup of the remote client, specify the clients and the policy in the following format:

   UNIX: `/usr/openv/pdde/pdag/bin/seedutil -seed -sclient client_name -spolicy policy_name -dclient destination_client_name`

   Windows: `install_path\Veritas\pdde\seedutil -seed -sclient client_name -spolicy policy_name -dclient destination_client_name`

   Note: NetBackup treats long and short host names differently, so ensure that you use the client name as it appears in the policy that backs it up.

   See "NetBackup seedutil options" on page 85.

2. Repeat the command for each client that you want to seed with fingerprints.
3. Verify that the seeding directories for the clients were created by using the following command:

   `seedutil -list_clients`

4. Back up the clients.
5. After the client or clients are backed up, remove the seeding directories for the clients. The following is the command syntax:

   `seedutil -clear client_name`

   After one full backup for the client or clients, NetBackup clears the seeding directory automatically. If the first backup fails, the seeded data remains for successive attempts. Although NetBackup clears the seeding directory automatically, Cohesity recommends that you clear the client seeding directories manually.

NetBackup seedutil options

The following is the usage statement for the `seedutil` utility:

`seedutil [-v log_level] [-seed -sclient source_client_name -spolicy policy_name -dclient destination_client_name [-backupid backup_id]] [-clear client_name] [-clear_all] [-list_clients] [-list_images client_name] [-dsid] [-help]`

The following items describe the options:

`-backupid backup_id`	The backup ID from which to copy the data for seeding.
`-clear client_name`	Clear the contents of the seed directory specified by the `client_name` .
`-clear_all`	Clear the contents of all of the seed directories.
`-dclient destination_client_name`	The name of the new client for which you are seeding the data.
`-dsid`	Data selection ID.
`-help`	Display help for the command.
`-list_clients`	List all of the clients that have been configured for seeding.
`-list_images client_name`	List the contents of the seeding directory for the specified client.
`-sclient source_client_name`	The client from which to copy the data for seeding. **Note:** NetBackup treats long and short host names differently, so ensure that you use the client name as it appears in the policy that backs it up.
`-seed`	Configure seeding.
`-spolicy policy_name`	The NetBackup policy that backed up the client that you want to use for the seeding data.
`-v log_level`	The log level.

The following are the directories in which the command resides:

- UNIX: `/usr/openv/pdde/pdag/bin`
- Windows: `C:\Program Files\Veritas\pdde`

About sampling and predictive cache

MSDP uses a memory up to a size that is configured in `MaxCacheSize` to cache fingerprints for efficient deduplication lookup. A new fingerprint cache lookup data scheme that is introduced in NetBackup release 10.1 reduces the memory usage. It splits the current memory cache into two components, sampling cache (S-cache)

and predictive cache (P-cache). S-cache caches a percentage of the fingerprints from each backup and is used to find similar data from the samples of previous backups for deduplication. P-cache caches the fingerprints that are most likely used in the immediate future for deduplication lookup.

At the start of a job, a small portion of the fingerprints from its last backup is loaded into P-cache as initial seeding. The fingerprint lookup is done with P-cache to find duplicates, and the lookup misses are searched from S-cache samples to find the possible matches of previous backup data. If found, part of the matched backup fingerprints is loaded into P-cache for future deduplication.

The S-cache and P-cache fingerprint lookup method is enabled for local and cloud storage volumes with MSDP non-BYO deployments including Flex, Flex Worm, Flex Scale, NetBackup Appliance, AKS, and EKS deployment. This method is also enabled for cloud-only volumes for MSDP BYO platforms. For the platforms with cloud-only volume support, local volume still uses the original cache lookup method. You can find S-cache and P-cache configuration parameters under the `Cache` section of the configuration file `contentrouter.cfg`.

From NetBackup 10.2, S-cache and P-cache fingerprint lookup method for local storage is used with the new setup for Flex, Flex WORM, and NetBackup Appliance. The upgrade does not change the S-cache and P-cache fingerprint lookup method.

The default values for S-cache and P-cache:

Configuration	Default value
`MaxCacheSize`	512MiB
`MaxPredictiveCacheSize`	40%
`MaxSamplingCacheSize`	20%
`EnableLocalPredictiveSamplingCache` under the `Cache` section in `contentrouter.cfg`	`true`
`EnableLocalPredictiveSamplingCache` under the `CA` section in `spa.cfg`	`true`

For the systems that use P/S cache, the local volume and cloud volumes share the same S-cache and P-cache size, and the overall memory is limited by `UsableMemoryLimit`.

The S-cache size is determined by the back-end MSDP capacity or the number of fingerprints from the back-end data. With the assumption that an average segment size of 32KB, the S-cache size is about 100MB per TB of back-end capacity. P-cache size is determined by the number of concurrent jobs and data locality or working

set of the incoming data. With working set of 250MB per stream (about 5 million fingerprints). For example, 100 concurrent streams need a minimum memory of 25GB (100*250MB). The working set can be larger for certain applications with multiple streams and large data sets. As P-cache is used for fingerprint deduplication lookup and all fingerprints that are loaded into P-cache stay there until its allocated capacity is reached, the larger the P-cache size, the better the potential lookup hit rate, and the more memory usage. Under-sizing S-cache or P-cache leads to reduced deduplication rates and over-sizing increases the memory cost.

Rebuilding the sampling cache

When MSDP is upgraded, the fingerprint lookup method does not change. If you want to use the new fingerprint lookup method predictive and sampling cache, you can configure it manually. Use the `rebuild_scache` script to rebuild the sampling cache during the maintenance window. It enhances the local deduplication ratio in post-upgrade environments when you enable predictive cache and sampling cache manually on local BYO, Flex, and Flex Worm environments.

The script is stored at the following location:

- UNIX: `/usr/openv/pdde/pdag/scripts/rebuild_scache.sh`
- Windows: `install_path\Veritas\pdde\rebuild_scache.bat`

You are not required to specify the log level. Logs are saved at the following location:

- UNIX: /var/log/puredisk/<date and time>-rebuild-scache.log
- Windows: C:\rebuild-scache.log

Consider the following recommendations when you run the script to rebuild the sampling cache.

- After the upgrade, run this script immediately before running the backup tasks to rebuild the sampling cache.
- Ensure that the predictive and sampling cache are enabled when the script is running.
- For S3 interface for MSDP and universal share environments, rebuilding the sampling cache may affect deduplication ratios.
- The script execution affects all backup jobs causing them to fail during the rebuilding process and the backup jobs are not performed.
  If the backup job cannot be performed after the rebuilding process is complete, restart the MSDP.
- When the script is initiated, ensure that it is successful. If it is terminated for some reason, run the script again.

- If errors occur during the rebuilding process, run the script again.

To rebuild the sampling cache after an upgrade

1. Enable predictive and sampling cache in `contentrouter.cfg` and `spa.cfg`.

   See "About sampling and predictive cache" on page 86.

2. On the storage server, run the script:

   UNIX:`/usr/openv/pdde/pdag/scripts/rebuild_scache.sh`

   Windows:`install_path\Program Files\Veritas\bin\rebuild_scache.bat`

3. Type `y` and press **Enter** to proceed. A warning about the job status is displayed.
4. Type `y` and press **Enter** to initiate the process to rebuild. The script displays the rebuild state and percentage.

In the Flex worm environment, use deduplication shell to start the sampling cache rebuilding.

To rebuild the sampling cache from the deduplication shell

1. Open an SSH session to the server.
2. Run the following command for the initial sampling cache rebuilding.

   ```
   setting rebuild-scache rebuild-sampling-cache
   ```

3. See the sampling cache rebuilding status.

   ```
   setting rebuild-scache rebuild-sampling-cache-status
   ```

Enabling 400 TB support for MSDP

Before you configure a storage server for a 400 TB **Media Server Deduplication Pool**, you must enable support for the multiple volumes that are required.

See "About MSDP capacity support and hardware requirements" on page 28.

See "About provisioning the storage for MSDP" on page 57.

Creating the data directories for 400 TB MSDP support

NetBackup requires that each storage volume contain a directory named `data`.

You must create the `data` directories on the second and third volumes that are required for 400 TB support. (NetBackup creates the required `data` directory on the volume that you specify in the **Storage server options UI**.)

Prerequisite

- The volumes must be formatted with the file systems that NetBackup supports for MSDP and mounted on the storage server.
  See "About provisioning the storage for MSDP" on page 57.
- The storage server must be configured already.
  See "Configuring a storage server for a Media Server Deduplication Pool" on page 93.

To create the data directories for 400 TB MSDP support

- In both the second and third volumes for the **Media Server Deduplication Pool**, create a `data` subdirectory at the volume's mount points, as follows:

  mount_point/data

  The following is an example of the mount points for the three required storage volumes:

```
/msdp/vol0  <--- Netbackup creates the data directory in this volume
/msdp/vol1  <--- Create a data directory in this volume
/msdp/vol2  <--- Create a data directory in this volume
```

Adding volumes to a 400 TB Media Server Deduplication Pool

When you configure a storage server for a 400 TB **Media Server Deduplication Pool**, you specify the pathname of the first storage volume. Before you can use the **Media Server Deduplication Pool**, you must add the other two volumes to the disk pool.

The following are the minimum hardware requirements for adding volumes to a 400 TB MSDP:

- **CPU:** A 64-bit processor with a minimum clock rate of 2.4-GHz is required. A minimum of 8 cores are required, 16 cores are recommended.
- **Memory:** Minimum 256 GB. You may need to add more memory if you have additional roles performed by the same media server. For example, when the media server is used as a VMware backup host, an NDMP backup agent and a primary server.
- **Swap:** 64 GB
- **Storage:**
  - **Metadata disk:** RAID 0+1 is recommended, with at least 1 TB of space.
  - Cohesity recommends eight mount points, each mount point must have a separate RAID group, a RAID 6 is recommended. Both the metadata disk and data disk should have more than 250 MB/sec of read or write speed.

- **File Systems:** NetBackup supports VxFS, XFS, or Ext4, but VxFS is recommended. The number of storage volumes can vary based on your setup. The maximum amount of storage space is 400 TB. The following procedure uses 8 file systems of 50 TB each for the example.

See "About provisioning the storage for MSDP" on page 57.

See "Configuring a storage server for a Media Server Deduplication Pool" on page 93.

To add volumes to a 400 TB Media Server Deduplication Pool

1. On the MSDP storage server, you must create, format, and mount new storage volumes. One of the storage volumes must have 1 TB or greater of storage space (this storage is for metadata). The other storage volumes can equal up to 400 TB of storage space.

   This procedure uses 8 file systems of 50 TB each for the example.

   Note: The number of storage volumes can vary based on your setup. The maximum amount of storage space is 400 TB.

2. Mount a 1 TB storage volume (for metadata) at:

   ```
   /msdp/cat
   ```

3. Mount the eight storage volumes on:

   ```
   /msdp/vol1
   ...
   /msdp/vol8
   ```

4. Create a touch a file at `/etc/nbapp-release` if it doesn't exist.

5. Create a subdirectory that is called `data` under each mounted volume:

   ```
   /msdp/vol1/data
   ...
   /msdp/vol8/data
   ```

6. On the left, click **Storage** > **Disk storage**.Configure the MSDP through the **Storage server options**UI. Enter alternate path for the deduplication database in the **Use alternate path for deduplication database** field.

7. Provide the storage path as `/msdp/vol1` and the database path as `/msdp/cat`.

8 Add additional 50 TB storage volumes to the deduplication pool:

```
/usr/openv/pdde/pdcr/bin/crcontrol --dsaddpartition /msdp/vol2/data
...
/usr/openv/pdde/pdcr/bin/crcontrol --dsaddpartition /msdp/vol8/data
```

If a non-root user is used for the NetBackup media server, run the following command to change the owner of the newly created volumes to the NetBackup media server service user:

```
chown -R <NBU-service-user>:root <MSDP-volume-path>
```

9 Verify that the deduplication pool contains the new volumes using the following command:

```
/usr/openv/pdde/pdcr/bin/crcontrol --dsstat 2 | grep Mount
Mount point count: 8
```

Configuring 400 TB on a Windows MSDP server

To enable 400 TB support, create the following file:

```
mkdir c:\etc
echo Windows_BYO > "c:\\etc\\nbapp-release"
```

The sizing recommendations for Windows are the same as they are for Linux. One of the storage volumes must have 1 TB of storage space and the other storage volumes can equal up to 400 TB of storage space. Using Windows, there are a few additional requirements:

- The `DCHeaderHashSize` setting in the `<MSDP Storage DIR>\etc\puredisk\contentrouter.cfg` file must be modified to be `2000000 / number_of_volumes`. For example, with the full eight mount points, set the `DCHeaderHashSize` to 250000.
- The volume that is used should be present as nested volumes, not as letter drives (C: Or E:). Cohesity qualified this solution using NTFS volumes.

The following is an example volume layout and each `data#` directory is a nested mount:

```
"msdp_data" :["f:/msdp/data1" ,"f:/msdp/data2" ,"f:/msdp/data3" ,
"f:/msdp/data4","f:/msdp/data5","f:/msdp/data6", "f:/msdp/data7"],
"f:/msdp/data8"],
"msdp_cat" :["f:/msdp/cat" ]
```

The `crcontrol` syntax is the same as Linux. On Windows, `crcontrol` is located in `<INSTALL_DRIVE>\Program Files\Veritas\pdde\`. For example:

```
C:\Program Files\Veritas\pdde\crcontrol --dsaddpartition f:\msdp\data2
```

Note: MSDP storage capacity has a defined maximum and not following these settings can result in performance-related issues due to data not being balanced across all volumes.

For more information about MSDP storage capacity review the following section:

See "About MSDP capacity support and hardware requirements" on page 28.

Note: NetBackup supports a pool size up to 400 TB. A pool can be a smaller size and expanded later by adding additional volumes.

Configuring a storage server for a Media Server Deduplication Pool

Configure in this context means to configure a NetBackup media server as a storage server for a **Media Server Deduplication Pool**.

See "About MSDP storage servers" on page 35.

The type of storage.	Select **Media Server Deduplication Pool** for the type of disk storage.
The credentials for the deduplication engine.	See "About the NetBackup Deduplication Engine credentials" on page 41.
The storage paths.	See "MSDP storage path properties" on page 95.
The network interface.	See "About the network interface for MSDP" on page 42.
The load-balancing servers, if any.	See "About MSDP storage servers" on page 35.

When you configure the storage server, the UI also lets you create a disk pool and storage unit.

Prerequisite	For a 96-TB **Media Server Deduplication Pool**, you must create the required directories before you configure the storage server. See "Creating the data directories for 400 TB MSDP support" on page 89.

To configure a NetBackup storage server for a Media Server Deduplication Pool

1. Open the web UI.
2. On the left, click **Storage > Disk storage**. Select the **Storage servers** tab, then click **Add**.
3. In the **Storage type** list, select **Disk storage servers**.
4. From the **Category options**, select **Media Server Deduplication Pool (MSDP, MSDP Cloud, MVG)**.
5. Click **Start**.
6. On the **Basic properties** tab, select or enter the appropriate information.

Media server	Select the media server that you want to configure as the storage server. You can add deduplication load-balancing servers from the UI.
Username	Enter the username for the NetBackup Deduplication Engine. See "About the NetBackup Deduplication Engine credentials" on page 41.
Password	Enter the password for the NetBackup Deduplication Engine.
Re- enter password	To confirm the password, re-enter the password.

7. On the **Storage server options** page, search or enter the **Storage path**.
8. Enter the alternate path in the **Use alternate path for deduplication database** field.
9. In **Use specific network interface** field, enter the interface.
10. If required, select the **Enable encryption** check box.
11. Click **Next**.
12. On the **Media servers** page, click **Add**.
13. Select the additional media servers.
14. Click **Add**.
15. Click **Next**.
16. On the **Review** page, verify all the information and click **Save**.

MSDP storage path properties

NetBackup requires that the storage is exposed as a directory path. The following table describes the storage path properties for a **Media Server Deduplication Pool** on the storage server:

Table 4-15 MSDP storage path properties

Property	Description
Storage path	The path to the storage. The storage path is the directory in which NetBackup stores the raw backup data. Backup data should not be stored on the system disk. Because the storage requires a directory path, do not use only the root node (`/`) or drive letter (`E:\`) as the storage path. (That is, do not mount the storage as a root node (`/`) or a drive letter (`E:\`). For a 400 TB **Media Server Deduplication Pool**, you must enter the path name of the mount point for the volume that you consider the first 32 TB storage volume. The following is an example of a volume naming convention for the mount points for the backups: `/msdp/vol0  <--- The first volume` `/msdp/vol1` `/msdp/vol2` NetBackup supports 400 TB deduplication pools on a subset of supported systems. See the *NetBackup Deduplication Guide*. See "About MSDP capacity support and hardware requirements" on page 28. See "About provisioning the storage for MSDP" on page 57. See "Creating the data directories for 400 TB MSDP support" on page 89. You can use the following characters in the storage path name: ■ Any of the 26 letters of the International Standards Organization (ISO) Latin-script alphabet, both uppercase (capital) letters and lowercase (small) letters. These are the same letters as the English alphabet. ■ Any integer from 0 to 9, inclusive. ■ A space character. ■ Any of the following characters: UNIX: `_-:./\` Windows: `_-:.\` (a colon (`:`) is allowed only after a drive letter (for example, `G:\MSDP_Storage`) NetBackup requirements for the deduplication storage paths may affect how you expose the storage. See the *NetBackup Deduplication Guide*. See "About MSDP storage and connectivity requirements" on page 30.

Table 4-15 MSDP storage path properties *(continued)*

Property	Description
Use alternate path for deduplication database	By default, NetBackup uses the storage path for the MSDP database (that is, the MSDP catalog) location. The MSDP database is different than the NetBackup catalog. Select this option to use a location other than the default for the deduplication database. For a 400 TB **Media Server Deduplication Pool**, you must select this option. See "About provisioning the storage for MSDP" on page 57. See the *NetBackup Deduplication Guide*. For performance optimization, it is recommended that you use a separate disk volume for the deduplication database than for the backup data.
Database path	If you selected **Use alternate path for deduplication database**, enter the path name for the database. The database should not be stored on the system disk. For a 400 TB **Media Server Deduplication Pool**, you must enter the path name of the partition that you created for the MSDP catalog. For example, if the naming convention for your mount points is `/msdp/volx` , the following path is recommended for the MSDP catalog directory: `/msdp/cat` See the *NetBackup Deduplication Guide*. See "About provisioning the storage for MSDP" on page 57. For performance optimization, it is recommended that you use a separate disk volume for the deduplication database than for the backup data. You can use the following characters in the path name: ■ Any of the 26 letters of the International Standards Organization (ISO) Latin-script alphabet, both uppercase (capital) letters and lowercase (small) letters. These are the same letters as the English alphabet. ■ Any integer from 0 to 9, inclusive. ■ A space character. ■ Any of the following characters: UNIX: `_-:./\` Windows: `_-:.\` (a colon (`:`) is allowed only after a drive letter (for example, `F:\MSDP_Storage`)

If the directory or directories do not exist, NetBackup creates them and populates them with the necessary subdirectory structure. If the directory or directories exist, NetBackup populates them with the necessary subdirectory structure.

Caution: You cannot change the paths after NetBackup configures the deduplication storage server. Therefore, decide during the planning phase where and how you want the deduplicated backup data to be stored and then carefully enter the paths.

See "Planning your MSDP deployment" on page 25.

See "About MSDP storage servers" on page 35.

See "About the NetBackup deduplication destination" on page 27.

MSDP network interface properties

The following table describes the network interface properties for a **Media Server Deduplication Pool** storage server.

Caution: You cannot change the network interface after NetBackup configures the deduplication storage server. Therefore, enter the properties carefully.

Table 4-16 MSDP network interface properties

Property	Description
Use specific network interface	Select this option to specify a network interface for the deduplication traffic. If you do not specify a network interface, NetBackup uses the operating system host name value. See "About the network interface for MSDP" on page 42.
Interface	If you selected **Use specific network interface**, enter the interface name.

About disk pools for NetBackup deduplication

NetBackup deduplication disk pools represent the storage for deduplicated backup data. NetBackup servers or NetBackup clients deduplicate the backup data that is stored in a deduplication disk pool.

Two types of deduplication pools exist, as follows:

- A NetBackup **Media Server Deduplication Pool** represents the disk storage that is attached to a NetBackup media server. NetBackup deduplicates the data and hosts the storage.
  NetBackup requires exclusive ownership of the disk resources that comprise the deduplication pool. If you share those resources with other users, NetBackup cannot manage deduplication pool capacity or storage lifecycle policies correctly.

How many deduplication pools you configure depends on your storage requirements. It also depends on whether or not you use optimized duplication or replication, as described in the following table:

Table 4-17 Deduplication pools for duplication or replication

Type	Requirements
Optimized duplication within the same NetBackup domain	Optimized duplication in the same domain requires the following deduplication pools: ■ At least one for the backup storage, which is the source for the duplication operations. The source deduplication pool is in one deduplication node. ■ Another to store the copies of the backup images, which is the target for the duplication operations. The target deduplication pool is in a different deduplication node. See "About MSDP optimized duplication within the same domain" on page 122.
Auto Image Replication to a different NetBackup domain	Auto Image Replication deduplication pools can be either replication source or replication target. The replication properties denote the purpose of the deduplication pool. The deduplication pools inherit the replication properties from their volumes. See "About the replication topology for Auto Image Replication" on page 143. Auto Image Replication requires the following deduplication pools: ■ At least one replication source deduplication pool in the originating domain. A replication source deduplication pool is one to which you send your backups. The backup images on the source deduplication pool are replicated to a deduplication pool in the remote domain or domains. ■ At least one replication target deduplication pool in a remote domain or domains. A replication target deduplication pool is the target for the duplication operations that run in the originating domain. See "About NetBackup Auto Image Replication" on page 139.

See "Changing a Media Server Deduplication Pool properties" on page 544.

See "Setting a Media Server Deduplication Pool attribute" on page 543.

Configuring a disk pool for deduplication

Before you can configure a NetBackup disk pool, a NetBackup deduplication storage server must exist.

See "About disk pools for NetBackup deduplication" on page 97.

When you configure a deduplication disk pool, you specify the following:

- The deduplication storage server to query for the disk storage to use for the pool.

- The disk volume to include in the pool.
  NetBackup exposes the storage as a single volume.
- The disk pool properties.

Cohesity recommends that disk pool names be unique across your enterprise.

To configure a deduplication disk pool

1. On the left, click **Storage > Disk storage**. Click the **Disk pools** tab, then click **Add**.

   Another way to create a disk pool is to click **Create disk pool** at the top of the screen after you have created a storage server.

2. In the **Disk pool options**, enter all the required information.

   Click **Change** to select a storage server.

   If the option **Limit I/O streams** is cleared, the default value is **Unlimited** and may cause performance issues.

   Click **Next**.

3. In **Volumes**, use the **Volume** drop down to select a volume or add a new volume. If you want to add a new disk pool volume, use the **Add volume** option.

   You can configure an MSDP storage server to use cloud storage. Select an existing cloud volume or create a new volume for the MSDP storage server.

   Note: When you enable Server-Side Encryption, you can configure AWS Customer-Managed keys. These keys cannot be deleted once they are in use by NetBackup. Each object is encrypted with the key during upload and deleting the key from AWS causes NetBackup restore failures.

   Note: FortKnox supports multiple options. For FortKnox Amazon and Amazon Government options and FortKnox Azure and Azure Government. In the web UI, you must contact your Cohesity NetBackup account manager for credentials or with any questions.

   For more information on environments and deployment, refer to FortKnox for NetBackup.

   For more information about FortKnox Amazon and FortKnox Azure options, refer to the NetBackup Deduplication Guide.

   Enter all the required information based on the selection and click **Next**.

4 In **Replication**, click **Add** to add replication targets to the disk pool.

This step lets you select a trusted primary server or add a trusted primary server. You can add a primary server that supports NetBackup Certificate Authority (NBCA), ECA, and ECA together with NBCA.

Replication is supported only on MSDP.

Review all the information that is entered for the replication targets and then click **Next**.

5 On the **Review** page, verify that all settings and information are correct. Click **Finish**.

The disk pool creation and replication configuration continue in the background if you close the window. If there is an issue with validating the credentials and configuration of the replication, you can use the **Change** option to adjust any settings.

Note: The image sharing server, which already has a disk pool that is created for it, is not available while NetBackup creates a new disk pool for image sharing.

See "Viewing Media Server Deduplication Pool attributes" on page 542.

Media Server Deduplication Pool properties

Table 4-18 describes the disk pool properties.

Table 4-18 Media server deduplication pool properties

Property	Description
Storage server	The storage server name. The storage server is the same as the NetBackup media server to which the storage is attached.
Storage server type	For a **Media Server Deduplication Pool**, the storage type is **PureDisk**.
Disk volumes	For a **Media Server Deduplication Pool**, all disk storage is exposed as a single volume. **PureDiskVolume** is a virtual name for the storage that is contained within the directories you specified for the storage path and the database path.
Total available space	The amount of space available in the disk pool.
Total raw size	The total raw size of the storage in the disk pool.

Table 4-18 Media server deduplication pool properties *(continued)*

Property	Description
Disk Pool name	The disk pool name. Enter a name that is unique across your enterprise.
Comments	A comment that is associated with the disk pool.
High water mark	The **High water mark** indicates that the volume is full. When the volume reaches the **High water mark**, NetBackup fails any backup jobs that are assigned to the storage unit. NetBackup also does not assign new jobs to a storage unit in which the deduplication pool is full. The **High water mark** includes the space that is committed to other jobs but not already used. The default is 98%.
Low water mark	The **Low water mark** has no affect on the **PureDiskVolume**.
Limit I/O streams	Select to limit the number of read and write streams (that is, jobs) for each volume in the disk pool. A job may read backup images or write backup images. By default, there is no limit. If you select this property, also configure the number of streams to allow per volume. When the limit is reached, NetBackup chooses another volume for write operations, if available. If not available, NetBackup queues jobs until a volume is available. Too many streams may degrade performance because of disk thrashing. Disk thrashing is excessive swapping of data between RAM and a hard disk drive. Fewer streams can improve throughput, which may increase the number of jobs that complete in a specific time period.
per volume	Select or enter the number of read and write streams to allow per volume. Many factors affect the optimal number of streams. Factors include but are not limited to disk speed, CPU speed, and the amount of memory.

See “About NetBackup media server deduplication” on page 34.

See “About disk pools for NetBackup deduplication” on page 97.

See “Configuring a disk pool for deduplication” on page 98.

See “Managing Media Server Deduplication Pools” on page 541.

See “About the NetBackup deduplication destination” on page 27.

Configuring a Media Server Deduplication Pool storage unit

A NetBackup deduplication storage unit represents the storage in either a **Media Server Deduplication Pool**. Create one or more storage units that reference the disk pool.

See “About disk pools for NetBackup deduplication” on page 97.

You may have created a storage unit when you created a disk pool. To determine if storage units exist for the disk pool, see the **Storage > Storage units**.

To configure a storage unit

1. Open the NetBackup web UI.
2. On the left, click **Storage > Storage units**.
3. Click **Add**.
4. Follow the instructions in the UI.

   For a storage unit for optimized duplication destination, select **Only use the following media servers**. Then select the media servers that are common between the two deduplication nodes.

   See “Media Server Deduplication Pool storage unit properties” on page 102.

Media Server Deduplication Pool storage unit properties

The following are the configuration options for a storage unit that has a **Media Server Deduplication Pool** as a target.

Table 4-19 Media Server Deduplication Pool storage unit properties

Property	Description
Storage unit name	A unique name for the new storage unit. The name can describe the type of storage. The storage unit name is the name used to specify a storage unit for policies and schedules. The storage unit name cannot be changed after creation.
Storage unit type	Select **Disk** as the storage unit type.
Disk type	Select **PureDisk** for the disk type for a **Media Server Deduplication Pool**.

Table 4-19 Media Server Deduplication Pool storage unit properties *(continued)*

Property	Description
Disk pool	Select the disk pool that contains the storage for this storage unit. All disk pools of the specified **Disk type** appear in the **Disk pool** list. If no disk pools are configured, no disk pools appear in the list.
Media server	The **Media server** setting specifies the NetBackup media servers that can deduplicate the data for this storage unit. Only the deduplication storage server and the load balancing servers appear in the media server list. Specify the media server or servers as follows: ■ To allow any server in the media server list to deduplicate data, select **Use any available media server**. ■ To use specific media servers to deduplicate the data, select **Only use the following media servers**. Then, select the media servers to allow. NetBackup selects the media server to use when the policy runs.
Maximum fragment size	For normal backups, NetBackup breaks each backup image into fragments so it does not exceed the maximum file size that the file system allows. You can enter a value from 20 MBs to 51200 MBs. For a FlashBackup policy, Cohesity recommends that you use the default, maximum fragment size to ensure optimal deduplication performance. For more information, see the *NetBackup NAS Administrator's Guide* and the *NetBackup Snapshot Manager for Data Center Administrator's Guide*.

Table 4-19 Media Server Deduplication Pool storage unit properties *(continued)*

Property	Description
Maximum concurrent jobs	The **Maximum concurrent jobs** setting specifies the maximum number of jobs that NetBackup can send to a disk storage unit at one time. (Default: one job. The job count can range from 0 to 256.) This setting corresponds to the Maximum concurrent write drives setting for a Media Manager storage unit. NetBackup queues jobs until the storage unit is available. If three backup jobs are scheduled and **Maximum concurrent jobs** is set to two, NetBackup starts the first two jobs and queues the third job. If a job contains multiple copies, each copy applies toward the **Maximum concurrent jobs** count. **Maximum concurrent jobs** controls the traffic for backup and duplication jobs but not restore jobs. The count applies to all servers in the storage unit, not per server. If you select multiple media servers in the storage unit and 1 for **Maximum concurrent jobs**, only one job runs at a time. The number to enter depends on the available disk space and the server's ability to run multiple backup processes. **Warning:** A Maximum concurrent jobs setting of 0 disables the storage unit.
Use WORM	This option is enabled for storage units that are WORM capable. WORM is the acronym for Write Once Read Many. Select this option if you want the backup images on this storage unit to be immutable and indelible until the WORM Unlock Time.

See "About storage unit groups for MSDP" on page 54.

See "Configuring a Media Server Deduplication Pool storage unit" on page 102.

MSDP storage unit recommendations

You can use storage unit properties to control how NetBackup performs, as follows:

Configure a favorable client-to-server ratio

Throttle traffic to the media servers

Configure a favorable client-to-server ratio

For a favorable client-to-server ratio, you can use one disk pool and configure multiple storage units to separate your backup traffic. Because all storage units use the same disk pool, you do not have to partition the storage.

For example, assume that you have 100 important clients, 500 regular clients, and four media servers. You can use two media servers to back up your most important clients and two media servers to back up you regular clients.

The following example describes how to configure a favorable client-to-server ratio:

- Configure the media servers for NetBackup deduplication and configure the storage.
- Configure a disk pool.
- Configure a storage unit for your most important clients (such as STU-GOLD). Select the disk pool. Select **Only use the following media servers**. Select two media servers to use for your important backups.
- Create a backup policy for the 100 important clients and select the STU-GOLD storage unit. The media servers that are specified in the storage unit move the client data to the deduplication storage server.
- Configure another storage unit (such as STU-SILVER). Select the same disk pool. Select **Only use the following media servers**. Select the other two media servers.
- Configure a backup policy for the 500 regular clients and select the STU-SILVER storage unit. The media servers that are specified in the storage unit move the client data to the deduplication storage server.

Backup traffic is routed to the wanted data movers by the storage unit settings.

Note: NetBackup uses storage units for media server selection for write activity (backups and duplications) only. For restores, NetBackup chooses among all media servers that can access the disk pool.

See "About NetBackup media server deduplication" on page 34.

See "Configuring a Media Server Deduplication Pool storage unit" on page 102.

Throttle traffic to the media servers

You can use the **Maximum concurrent jobs** settings on disk pool storage units to throttle the traffic to the media servers. Effectively, this setting also directs higher loads to specific media servers when you use multiple storage units for the same disk pool. A higher number of concurrent jobs means that the disk can be busier than if the number is lower.

For example, two storage units use the same set of media servers. One of the storage units (STU-GOLD) has a higher **Maximum concurrent jobs** setting than the other (STU-SILVER). More client backups occur for the storage unit with the higher **Maximum concurrent jobs** setting.

See "Configuring a Media Server Deduplication Pool storage unit" on page 102.

Configuring client attributes for MSDP client-side deduplication

To configure client deduplication, set an attribute in the NetBackup primary server **Client attributes** host properties. If the client is in a backup policy in which the storage destination is a **Media Server Deduplication Pool**, the client deduplicates its own data.

To specify the clients that deduplicate backups

1. Open the NetBackup web UI.
2. On the left, click **Host > Host properties**.
3. Select the primary server.
4. If necessary, click connect. Then click **Edit primary server**.
5. Click **Client attributes**.
6. On the **General** tab, add the clients that you want to deduplicate their own data to the **Clients** list:
   - Click **Add**.
   - Enter a client name or browse to select a client. Then click **Add**. Repeat for each client that you want to add.
7. To select a deduplication location for a specific client, select that client.
8. Select one of the following **Deduplication location** options:
   - **Always use the media server** - disables client deduplication. By default, all clients are configured with the **Always use the media server** option.
   - **Prefer to use client-side deduplication** - uses client deduplication if the deduplication plug-in is active on the client. If it is not active, a normal backup occurs; client deduplication does not occur.
   - **Always use client-side deduplication** - uses client deduplication. If the deduplication backup job fails, NetBackup retries the job.

   You can override the **Prefer to use client-side deduplication** or **Always use client-side deduplication** host property in the backup policies.

See **Client-side deduplication** in the NetBackup Administrator's Guide, Volume I.

See "Disabling MSDP client-side deduplication for a client" on page 107.

See "About NetBackup Client Direct deduplication" on page 38.

See "About the NetBackup deduplication options" on page 22.

Disabling MSDP client-side deduplication for a client

You can remove a client from the list of clients that deduplicate their own data. If you do so, a deduplication server backs up the client and deduplicates the data.

To disable MSDP client deduplication for a client

1. Open the web UI.
2. On the left, click **Host > Host properties**.
3. Select the primary server.
4. If necessary, click **Connect**. Then click **Edit primary server**.
5. Click **Client attributes**.
6. Select the client that deduplicates its own data.
7. From the **Deduplication** list, select **Always use the media server**.
8. Click **Save**.

Disable client-side deduplication for all clients in a policy

To disable the use of client-side deduplication for a policy, select **Disable for all clients** on the **Attributes** tab.

To disable client-side deduplication for all clients in a policy

1. Open the web UI.
2. Click **Protection > Policies**.
3. Click on the policy that you want to edit.
4. Click the **Attributes** tab.
5. Locate **Client-side deduplication** and click **Disable for all clients**.
6. Click **Save**.

About MSDP compression

NetBackup deduplication hosts provide compression for the deduplicated data. It is separate from and different than NetBackup policy-based compression.

Compression is configured by default on all MSDP hosts. Therefore, backups, duplication traffic, and replication traffic are compressed on all MSDP hosts. The data also is compressed on storage.

Table 4-20 describes the compression options.

A different topic describes the interaction of the encryption and the compression settings for MSDP.

Table 4-20 MSDP compression options

Option	Description
Compression for backups	For backups, the deduplication plug-in compresses the data after it is deduplicated. The data remains compressed during transfer from the plug-in to the NetBackup Deduplication Engine on the storage server. The Deduplication Engine writes the encrypted data to the storage. For restore jobs, the process functions in the reverse direction. The `COMPRESSION` parameter in the `pd.conf` file on each MSDP host controls compression and decompression for that host. By default, backup compression is enabled on all MSDP hosts. Therefore, compression and decompression occur on the following hosts as necessary: ■ The clients that deduplicate their own data (that is, client-side deduplication). ■ The load balancing servers. ■ The storage server. MSDP compression cannot occur on normal NetBackup clients (that is, the clients that do not deduplicate their own data). **Note:** Do not enable backup compression by selecting the **Compression** option on the **Attributes** tab of the **Policy** dialog box. If you do, NetBackup compresses the data before it reaches the plug-in that deduplicates it. Consequently, deduplication rates are very low. Also, NetBackup does not use the Deduplication Multi-Threaded Agent if policy-based encryption is configured. See "About the MSDP Deduplication Multi-Threaded Agent" on page 66.

Table 4-20 MSDP compression options *(continued)*

Option	Description
Compression for duplication and replication	For duplication and replication, the deduplication plug-in compresses the data for transfer. The data remains compressed during transfer from the plug-in to the NetBackup Deduplication Engine on the storage server and remains compressed on the storage. The `OPTDUP_COMPRESSION` parameter in the `pd.conf` file controls compression for duplication and replication. By default, duplication and replication compression is enabled on all MSDP hosts. Therefore, duplication and replication compression occurs on the following MSDP servers: ■ The load balancing servers. ■ The storage server. Duplication and replication compression does not apply to clients. NetBackup chooses the least busy host to initiate and manage each duplication job and replication job. To ensure that compression occurs for all optimized duplication and replication jobs: do not change the default setting of the `OPTDUP_COMPRESSION` parameter.

See "About the MSDP pd.conf configuration file" on page 178.

See "Use MSDP compression and encryption" on page 53.

About MSDP encryption

NetBackup provides encryption for the deduplicated data. It is separate from and different than NetBackup policy-based encryption.

NetBackup uses the Advanced Encryption Standard 256 bit, CTR (AES) encryption algorithm to Media Server Deduplication Pool (MSDP). The AES encryption algorithm replaces the older Blowfish encryption algorithm.

See "About the Encryption Crawler" on page 826.

See "MSDP encryption behavior and compatibilities" on page 113.

Configuring encryption for MSDP local storage volume

Encrypting a key with another key is called envelop encryption. MSDP uses an envelope encryption.

More information is available on how to configure and to use KMS.

See "About MSDP Encryption using NetBackup Key Management Server service" on page 114.

For MSDP initial setup, you can use NetBackup web UI to configure encryption. Use the following steps manually to enable encryption for the existing systems. Once enabled, all data to the MSDP server local disk volume including NetBackup media servers, servers in opt-dup, servers in AIR, and client direct hosts is encrypted. You are not required to configure encryption at any other places.

Note: The following steps are for MSDP local disk volume only. For MSDP cloud volume encryption, see the following topic.

See "Configuring encryption for MSDP cloud storage volumes" on page 110.

To configure backup encryption for MSDP local storage volume

1. On the storage server, open the `contentrouter.cfg` file in a text editor; it resides in the following directory:
   - (UNIX) *storage_path*`/etc/puredisk`
   - (Windows) *storage_path*`\etc\puredisk`
2. Add `encrypt` to the `ServerOptions` line of the file. For example:

   ```
   ServerOptions=fast,verify_data_read,encrypt
   ```

   Encryption is enabled for all the data that is stored on the server, which includes the MSDP storage server, the MSDP load-balancing servers, and the NetBackup Client Direct deduplication clients.
3. Restart the MSDP services.

Note: Encryption configuration using the `pd.conf` file needs changes in NetBackup media servers or clients, and its use is deprecated.

Configuring encryption for MSDP cloud storage volumes

MSDP cloud volume encryption is configured through the NetBackup web UI or command line options. At the time of MSDP cloud volume creation, you can configure an encryption. For cloud volumes, KMS is always required to configure an encryption. The encryption checkbox is not available if KMS is not enabled.

Note: Each MSDP storage volume encryption is configured individually. Like MSDP disk storage pool encryption, once it is configured, all data to the cloud storage volume will be encrypted regardless of the data source.

See "Creating a cloud storage unit" on page 299.

Configuring MSDP encryption on different platforms

MSDP can be deployed on different platforms such as NetBackup BYO, NetBackup Appliance, Flex media server, Flex WORM, Flex Scale, Cloud Scale, and Access Appliance. Some of the platforms are closed and the configuration files cannot be edited directly. In those cases, platform-specific support is required. For example, `CLISH` in NetBackup Appliance and `Deduplication Shell` for Flex WORM and Access Appliance can be used to make the configuration changes.

Modes of rolling data conversion

MSDP uses the rolling data conversion mechanism to convert Blowfish encrypted data to AES-256 encrypted data, and MD5-like fingerprints to SHA-512/256 fingerprints, in parallel. There are two modes of data conversion: normal mode and fast mode.

- Normal mode: By default, the data conversion process starts in a normal mode for an upgraded system. Similar to compaction, the data conversion runs only when no backup, restore, or Content Router Queue Processing (CRQP) jobs are active.

  In the normal mode, the time for data conversion depends on the following factors:

  - Total size of the storage
  - Power of the CPU
  - Workload on the system

  Data conversion in the normal mode may take a longer time.

  Cohesity tests in a controlled environment showed that for a single 1-TB mount point, the conversion speed is about 50MB/s in normal mode.

- Fast mode: In the fast mode, the data conversion disables cyclic redundancy checks and compaction. The rolling data conversion runs while backup, restore, duplication, or CRQP jobs are active.

  Cohesity tests in a controlled environment showed that for a single 1-TB mount point, the conversion speed is about 105MB/s in fast mode.

Note: The performance numbers shown were observed in the Cohesity test environment and are not a guarantee of performance in your environment.

In a new installation of NetBackup 8.1, the rolling data conversion is marked as **Finished** and does not start in the future. For an upgrade to NetBackup 8.1, the rolling data conversion is enabled by default and works in the background after the

MSDP conversion completes. Only the data that existed before upgrade is converted. All new data uses the new SHA-512/256 fingerprint and does not need conversion.

While in **Fast mode**, the rolling data conversion affects the performance of backup, restore, duplication, and replication jobs. To minimize this effect, use the **Normal mode**, which pauses the conversion when the system is busy, but slows down the conversion process. The **Fast mode** keeps the conversion active regardless of system state.

You can manage and monitor the rolling data conversion using the following `crcontrol` command options.

Table 4-21 MSDP `crcontrol` command options for rolling data conversion

Option	Description
`--dataconverton`	To start the data conversion process, use the `--dataconverton` option: Windows: `install_path\Veritas\pdde\Crcontrol.exe --dataconverton` UNIX: `/usr/openv/pdde/pdcr/bin/crcontrol --dataconverton`
`--dataconvertoff`	To stop the data conversion process, use the `--dataconvertoff` option: Windows: `install_path\Veritas\pdde\Crcontrol.exe --dataconvertoff` UNIX: `/usr/openv/pdde/pdcr/bin/crcontrol --dataconvertoff`
`--dataconvertstate`	To determine the mode of data conversion and the conversion progress, use the `--dataconvertstate` option: Windows: `install_path\Veritas\pdde\Crcontrol.exe --dataconvertstate` UNIX: `/usr/openv/pdde/pdcr/bin/crcontrol --dataconvertstate`

Table 4-21 MSDP `crcontrol` command options for rolling data conversion *(continued)*

Option	Description
`--dataconvertmode`	To switch between the normal mode and fast mode of data conversion, use the `--dataconvertmode` option: Windows: `install_path\Veritas\pdde\Crcontrol.exe --dataconvertmode mode` UNIX: `/usr/openv/pdde/pdcr/bin/crcontrol --dataconvertmode <mode>` The default value for the *<mode>* variable is 0, which stands for the normal mode. To switch data conversion from normal mode to fast mode, enter 1 for the value of the *<mode>*variable.

See "MSDP encryption behavior and compatibilities" on page 113.

MSDP encryption behavior and compatibilities

MSDP supports multiple encryption algorithms. Therefore, it manages both the Blowfish and the AES encrypted data to ensure data compatibility.

For restore operations, MSDP recognizes the Blowfish data and the AES data to be able to restore the old backup images.

The following tables describe the encryption behavior for backup, duplication, and replication operations when the encryption is in progress.

Table 4-22 Encryption behavior for a backup operation to a NetBackup 8.0 storage server

Type of client	Data encryption format
Client with NetBackup 8.0, including the Client Direct deduplication	AES
Client with NetBackup version earlier than 8.0, excluding Client Direct deduplication	AES
Client with NetBackup version earlier than 8.0, using the Client Direct deduplication	AES (using inline data conversion)
Load balancing server with NetBackup version 8.0	AES

Table 4-22 Encryption behavior for a backup operation to a NetBackup 8.0 storage server *(continued)*

Type of client	Data encryption format
Load balancing server with NetBackup version earlier than 8.0	AES (using inline data conversion)

Table 4-23 Encryption behavior for optimized duplication and Auto Image Replication operations to a NetBackup 8.0 target server

Type of source storage	Data encryption format for the duplication or the replication data that is encrypted with AES	Data encryption format for the duplication or the replication data that is encrypted with Blowfish
Source server with NetBackup 8.0	AES	AES (using inline data conversion)
Source server with NetBackup version earlier than 8.0	Not applicable	AES (using inline data conversion)

Note: Inline data conversion takes place simultaneously while the backup, duplication, or replication operations are in progress.

About MSDP Encryption using NetBackup Key Management Server service

NetBackup incorporates the Key Management Server (KMS) with Media Server Deduplication Pool.

Starting NetBackup 10.1.1 and Flex WORM Storage Server 17.1, MSDP uses envelope encryption with multiple layers of keys to encrypt the data. Each new MSDP data segment is encrypted with a unique data encryption key (DEK) that is generated by an MSDP. Each DEK is encrypted or wrapped by a key encryption key (KEK) that is generated by an MSDP. Multiple DEKs can use the same KEK for encryption until a new active KEK is generated. The KEKs are encrypted using the active root keys that reside in NetBackup KMS or an external KMS. The root keys are not sent to the MSDP. Instead, an MSDP sends the KEK to the NetBackup primary server, which interfaces with the NetBackup KMS or an external KMS to encrypt or decrypt the KEK. The encrypted DEKs and encrypted KEKs are stored within MSDP.

For information on how to convert the legacy KMS encryption to the KEK-based KMS encryption for the previous backup data, See "Converting the legacy KMS to KEK-based KMS" on page 846.

User manages KMS service to create and activate a key. In KMS service, one active key must exist.

You can configure the KMS service from the NetBackup web UI or the NetBackup command line during storage server configuration.

Note: You cannot disable the MSDP KMS service after you enable it.

If the KMS service is not available for MSDP or the key in the KMS service that MSDP uses is not available, then MSDP waits in an infinite loop and the backup job may fail. When MSDP goes in an infinite loop, some commands that you run might not respond.

After you configure KMS encryption or once the MSDP processes restart, check the KMS encryption status after the first backup finishes.

The keys in the key dictionary must not be deleted, deprecated, or terminated. All keys that are associated with the MSDP disk pool must be in an active or an inactive state.

You can use the following commands to get the status of the KMS mode:

- For UNIX:

  `/usr/openv/pdde/pdcr/bin/crcontrol --getmode`

  For MSDP cloud, run the following `keydictutil` command to check if the Logical Storage Unit (LSU) is in KMS mode:

  `/usr/openv/pdde/pdcr/bin/keydictutil --list`

- For Windows:

  `<install_path>\Veritas\pdde\crcontrol.exe --getmode`

Note: If you use the `nbdevconfig` command to add a new encrypted cloud LSU and an encrypted LSU exists in this MSDP, the `keygroupname` must be the same as the `keygroupname` in the previous encrypted LSU.

For enabling KMS, refer to the following topics:

See "Configuring a storage server for a Media Server Deduplication Pool" on page 93.

Enabling KMS encryption for a local LSU

You can enable the KMS encryption for a local LSU by using the `nbdevconfig` command.

To enable KMS encryption for a local LSU

1. Create a configuration file with a name of your preference in the following format. For example, `sample_config.txt`

```
V7.5 "operation" "set-local-lsu-kms-property" string
V7.5 "encryption" "1" string
V7.5 "kmsenabled" "1" string
V7.5 "kmsservertype" "0" string
V7.5 "kmsservername" "xxxxxx" string
V7.5 "keygroupname" "xxxxx" string
```

Configuration setting	Description
`V7.5 "operation" "set-local-lsu-kms-property" string`	You can only update the KMS status from disabled to enabled.
`V7.5 "encryption" "1" string`	Specifies the encryption status. This value must be 1.
`V7.5 "kmsenabled" "1" string`	Specifies the KMS status. This value must be 1.
`V7.5 "kmsservertype" "0" string`	Specifies the KMS server type. This value must be 0.
`V7.5 "kmsservername" "" string`	KMS server name that is shared among all LSUs.

Configuration setting	Description
`V7.5 "keygroupname" "" string`	The key group name must include the following valid characters. ■ A-Z ■ a-z ■ 0-9 ■ Underscore (_) ■ Hyphen (-) ■ Colon (:) ■ Period (.) ■ Space

2 Run the following command to enable the KMS encryption:

```
nbdevconfig -setconfig -storage_server <storage server host name>
-stype PureDisk -configlist <configuration file name>
```

Note: All encrypted LSUs in one storage server must use the same `keygroupname` and `kmsservername`. KMS server must be configured. Key group and Key exist in the KMS server.

Upgrading KMS for MSDP

Before you upgrade KMS encryption from NetBackup version earlier than 8.1.1, complete the following steps. During the NetBackup upgrade, KMS rolling conversion runs along with MSDP encryption rolling conversion.

For NetBackup version earlier than 8.1.1, the supported NetBackup upgrade paths are:

- NetBackup 8.0 to 8.1.1 or later
- NetBackup 8.1 to 8.1.1 or later

For additional information, refer to the *Configuring KMS* section in the *NetBackup Security and Encryption Guide*.

Before you upgrade KMS, complete the following steps:

Note: The following steps are not supported on Solaris OS. For Solaris, refer to the following article:

Upgrade KMS encryption for MSDP on the Solaris platform

1. Create an empty database using the following command:

   - For UNIX:

     ```
     /usr/openv/netbackup/bin/nbkms -createemptydb
     ```

   - For Windows:

     ```
     <install_path>\Veritas\NetBackup\bin\nbkms.exe -createemptydb
     ```

     Enter the following parameters when you receive a prompt:

     - Enter the HMK passphrase
       Enter a password that you want to set as the host master key (HMK) passphrase. Press Enter to use a randomly generated HMK passphrase. The passphrase is not displayed on the screen.
     - Enter HMK ID
       Enter a unique ID to associate with the host master key. This ID helps you to determine an HMK associated with any key store.
     - Enter KPK passphrase
       Enter a password that you want to set as the key protection key (KPK) passphrase. Press Enter to use a randomly generated HMK passphrase. The passphrase is not displayed on the screen.
     - Enter KPK ID
       Enter a unique ID to associate with the key protection key. This ID helps you to determine a KPK associated with any key store.

   After the operation completes successfully, run the following command on the primary server to start KMS:

   - For UNIX:

     ```
     /usr/openv/netbackup/bin/nbkms
     ```

   - For Windows:

     ```
     sc start NetBackup Key Management Service
     ```

2. Create a key group and an active key by entering the following commands:

   - For UNIX:

     ```
     /usr/openv/netbackup/bin/admincmd/nbkmsutil -createkg -kgname
     msdp
     /usr/openv/netbackup/bin/admincmd/nbkmsutil -createkey -kgname
     msdp -keyname name –activate
     ```

   - For Windows:

     ```
     <install_path>\Veritas\NetBackup\bin\admincmd\nbkmsutil.exe
     -createkg -kgname msdp
     ```

```
<install_path>\Veritas\NetBackup\bin\admincmd\nbkmsutil.exe
-createkey -kgname msdp -keyname name -activate
```

Enter a password that you want set as the key passphrase.

3 Create a `kms.cfg` configuration file at the following location on the NetBackup media server where you have configured the MSDP storage:

- On UNIX:
  `/usr/openv/pdde/kms.cfg`
- On Windows:
  `<install_path>\Veritas\pdde\kms.cfg`

Add the following content to the `kms.cfg` file:

```
[KMSOptions]
KMSEnable=true
KMSKeyGroupName=YourKMSKeyGroupName
KMSServerName=YourKMSServerName
KMSType=0
```

For `KMSServerName`, enter the hostname of the server where the KMS service runs, mainly the primary server host name.

After completing the steps, you can upgrade MSDP.

About MSDP Encryption using external KMS server

NetBackup supports keys from an external key management service (KMS) server that encrypts data in case of MSDP storage. Keys are retrieved from the external KMS server to the primary server to encrypt or decrypt the key encryption keys (KEK) from an MSDP.

For information about external KMS support, see the NetBackup Security and Encryption Guide.

The other information remains the same as mentioned in the following topic:

See “About MSDP Encryption using NetBackup Key Management Server service” on page 114.

Configuring optimized synthetic backups for MSDP

To configure optimized synthetic backups for MSDP, you must select the **Synthetic backup** policy attribute.

To configure optimized synthetic backups for MSDP

1. Configure a **Standard** or **MS-Windows** backup policy.

   See the *NetBackup Administrator's Guide, Volume I*:

   https://support.cohesity.com/s/article/article-100040135

2. Select the **Synthetic backup** attribute on the **ScheduleAttribute** tab of the backup policy.

See "Setting MSDP storage server attributes" on page 533.

About a separate network path for MSDP duplication and replication

You can use a different network for MSDP duplication and replication traffic rather than the one you use for MSDP backups. Both the duplication and the replication data traffic and the control traffic travel over the separate network. Your MSDP traffic uses two different networks, as follows:

Backups and restores	For backups and restores, NetBackup uses the network interface that was configured during the storage server configuration. Both the backup and restore traffic and the control traffic travel over the *backup* network. See "About the network interface for MSDP" on page 42.
Duplication and replication	For the duplication and the replication traffic, configure your host operating systems to use a different network than the one you use for backups and restores. Both the duplication and the replication data traffic and the control traffic travel over the *duplication and replication* network. See "Configuring a separate network path for MSDP duplication and replication" on page 121. When you configure the optimized duplication or replication target, ensure that you select the host name that represents the *duplication and replication* network.

See “About MSDP optimized duplication within the same domain” on page 122.

See “About MSDP replication to a different domain” on page 136.

Configuring a separate network path for MSDP duplication and replication

You can use a different network for MSDP duplication and replication traffic rather than the one you use for MSDP backups. Both the duplication and the replication data traffic and the control traffic travel over the separate network.

See “About a separate network path for MSDP duplication and replication” on page 120.

This procedure describes how to use the storage servers `hosts` files to route the traffic onto the separate network.

The following are the prerequisites:

- Both the source and the destination storage servers must have a network interface card that is dedicated to the other network.
- The separate network must be operational and using the dedicated network interface cards on the source and the destination storage servers.
- On UNIX MSDP storage servers, ensure that the Name Service Switch first checks the local `hosts` file for before querying the Domain Name System (DNS). See the operating system documentation for information about the Name Service Switch.

To configure a separate network path for MSDP duplication and replication

1. On the source storage server, add the destination storage servers's dedicated network interface to the operating system `hosts` file. If *TargetStorageServer* is the name of the destination host on the network that is dedicated for duplication, the following is an example of the `hosts` entry in IPv4 notation:

   ```
   10.10.10.1   TargetStorageServer.example.com   TargetStorageServer
   ```

   Cohesity recommends that you always use the fully qualified domain name when you specify hosts.

2. On the destination storage server, add the source storage servers's dedicated network interface to the operating system `hosts` file. If *SourceStorageServer* is the name of the source host on the network that is dedicated for duplication, the following is an example of the `hosts` entry in IPv4 notation:

   ```
   10.80.25.66   SourceStorageServer.example.com   SourceStorageServer
   ```

   Cohesity recommends that you always use the fully qualified domain name when specifying hosts.

3. To force the changes to take effect immediately, flush the DNS cache. See the operating system documentation for how to flush the DNS cache.

4. From each host, use the `ping` command to verify that each host resolves the name of the other host.

   ```
   SourceStorageServer.example.com> ping TargetStorageServer.example.com
   TargetStorageServer.example.com> ping SourceStorageServer.example.com
   ```

   If the `ping` command returns positive results, the hosts are configured for duplication and replication over the separate network.

5. When you configure the target storage server, ensure that you select the host name that represents the alternate network path.

About MSDP optimized duplication within the same domain

Optimized duplication within the same domain copies the deduplicated backup images between **Media Server Deduplication Pools** within the same domain. The source and the destination storage must use the same NetBackup primary server.

The optimized duplication operation is more efficient than normal duplication. Only the unique, deduplicated data segments are transferred. Optimized duplication reduces the amount of data that is transmitted over your network.

Optimized duplication is a good method to copy your backup images off-site for disaster recovery.

By default, NetBackup does not retry the failed optimized duplication jobs that NetBackup Vault invokes using the `bpduplicate` command. You can change that behavior.

See "Configuring NetBackup optimized duplication or replication behavior" on page 133.

You can use a separate network for the duplication traffic.

See "About a separate network path for MSDP duplication and replication" on page 120.

See "Configuring MSDP optimized duplication within the same NetBackup domain" on page 129.

Review the following requirements and limitations.

About MSDP optimized duplication requirements

The following are the requirements for optimized duplication within the same NetBackup domain:

- The source storage and the destination storage must have at least one media server in common.
  See "About the media servers for MSDP optimized duplication within the same domain" on page 124.
- In the storage unit you use for the destination for the optimized duplication, you must select only the common media server or media servers.
  If you select more than one, NetBackup assigns the duplication job to the least busy media server. If you select a media server or servers that are not in common, the optimized duplication job fails.
  For more information about media server load balancing, see the NetBackup Administrator's Guide, Volume I.
- The destination storage unit cannot be the same as the source storage unit.

About MSDP optimized duplication limitations

The following are limitations for optimized duplication within the same NetBackup domain:

- If an optimized duplication job fails after the configured number of retries, NetBackup does not run the job again.
  By default, NetBackup retries an optimized duplication job three times. You can change the number of retries.

See "Configuring NetBackup optimized duplication or replication behavior" on page 133.

- NetBackup does not support MSDP optimized duplication to storage unit groups. If you use a storage unit group as a destination for optimized duplication, NetBackup uses regular duplication.
- Optimized duplication does not support multiple copies. If NetBackup is configured to make multiple new copies from the (source) copy of the backup image, the following occurs:
  - In a storage lifecycle policy, one duplication job creates one optimized duplication copy. If multiple optimized duplication destinations exist, a separate job exists for each destination. This behavior assumes that the device for the optimized duplication destination is compatible with the device on which the source image resides.
    If multiple remaining copies are configured to go to devices that are not optimized duplication capable, NetBackup uses normal duplication. One duplication job creates those multiple copies.
  - For other duplication methods, NetBackup uses normal duplication. One duplication job creates all of the copies simultaneously. The other duplication methods include the following:
    NetBackup Vault, the `bpduplicate` command line, and the duplication option of the **Catalog** utility in the **NetBackup web UI**.
- The copy operation uses the maximum fragment size of the source storage unit, not the setting for the destination storage unit. The optimized duplication copies the image fragments as is. For greater efficiency, the duplication does not resize and reshuffle the images into a different set of fragments on the destination storage unit.

See "About the media servers for MSDP optimized duplication within the same domain" on page 124.

See "About MSDP push duplication within the same domain" on page 125.

See "About MSDP pull duplication within the same domain" on page 128.

See "About MSDP optimized duplication and replication" on page 45.

About the media servers for MSDP optimized duplication within the same domain

For optimized **Media Server Deduplication Pool** duplication within the same domain, the source storage and the destination storage must have at least one media server in common. The common server initiates, monitors, and verifies the duplication operation. The common server requires credentials for both the source

storage and the destination storage. (For deduplication, the credentials are for the NetBackup Deduplication Engine, not for the host on which it runs.)

Which media server initiates the duplication operation determines if it is a push or a pull operation, as follows:

- If the media server is co-located physically with the source storage server, it is push duplication.
- If the media server is co-located physically with the destination storage server, it is a pull duplication.

Technically, no advantage exists with a push duplication or a pull duplication. However, the media server that initiates the duplication operation also becomes the write host for the new image copies.

A storage server or a load balancing server can be the common server. The common server must have the credentials and the connectivity for both the source storage and the destination storage.

See “About MSDP optimized duplication within the same domain” on page 122.

See “About MSDP push duplication within the same domain” on page 125.

See “About MSDP pull duplication within the same domain” on page 128.

About MSDP push duplication within the same domain

Figure 4-1 shows a push configuration for optimized duplication within the same domain. The local deduplication node contains normal backups; the remote deduplication node is the destination for the optimized duplication copies. Load balancing server LB_L2 has credentials for both storage servers; it is the common server.

Figure 4-1 Push duplication environment

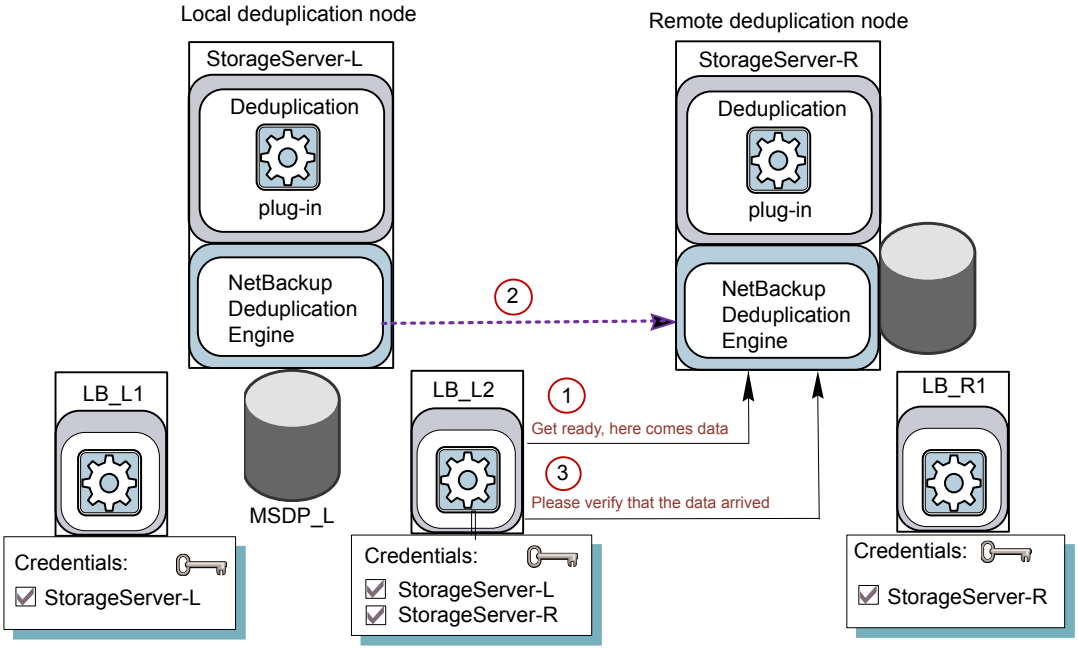


Figure 4-2 shows the settings for the storage unit for the normal backups for the local deduplication node. The disk pool is the **MSDP_L** in the local environment. Because all hosts in the local node are co-located, you can use any available media server for the backups.

Figure 4-2 Storage unit settings for backups to MSDP_L

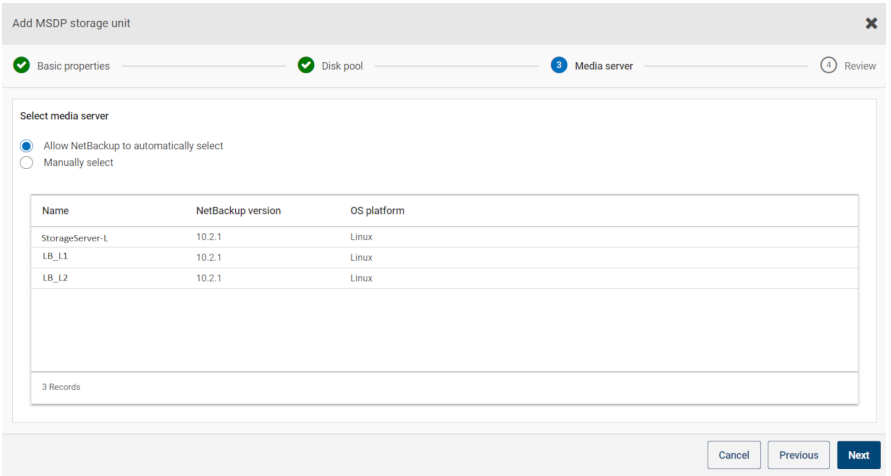
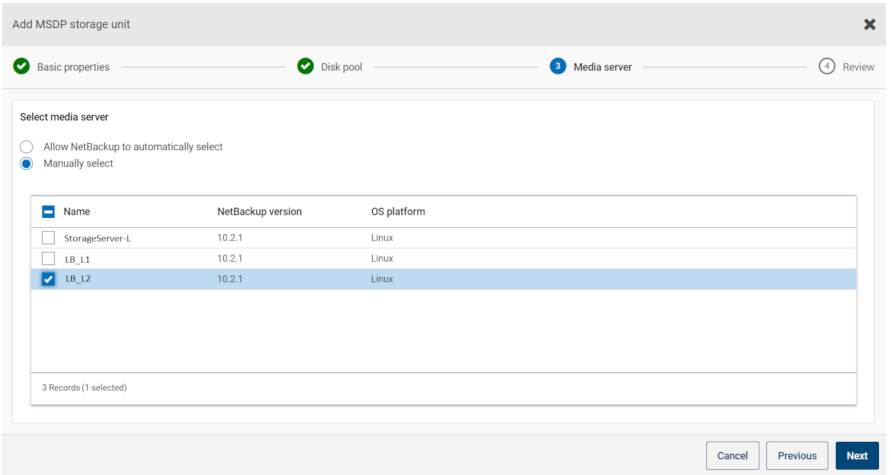


Figure 4-3 shows the storage unit settings for the optimized duplication. The destination is the **MSDP_R** in the remote environment. You must select the common server, so only the load balancing server LB_L2 is selected.

Figure 4-3 Storage unit settings for duplication to MSDP_R



If you use the remote node for backups also, select **StorageServer-R** and load balancing server **LB_R1** in the storage unit for the remote node backups. If you select server **LB_L2**, it becomes a load balancing server for the remote **Media Server Deduplication Pool**. In such a case, data travels across your WAN.

About MSDP pull duplication within the same domain

Figure 4-4 shows a pull configuration for optimized duplication within the same domain. Deduplication node A contains normal backups; deduplication node B is the destination for the optimized duplication copies. Host B has credentials for both nodes; it is the common server.

Figure 4-4 Pull duplication

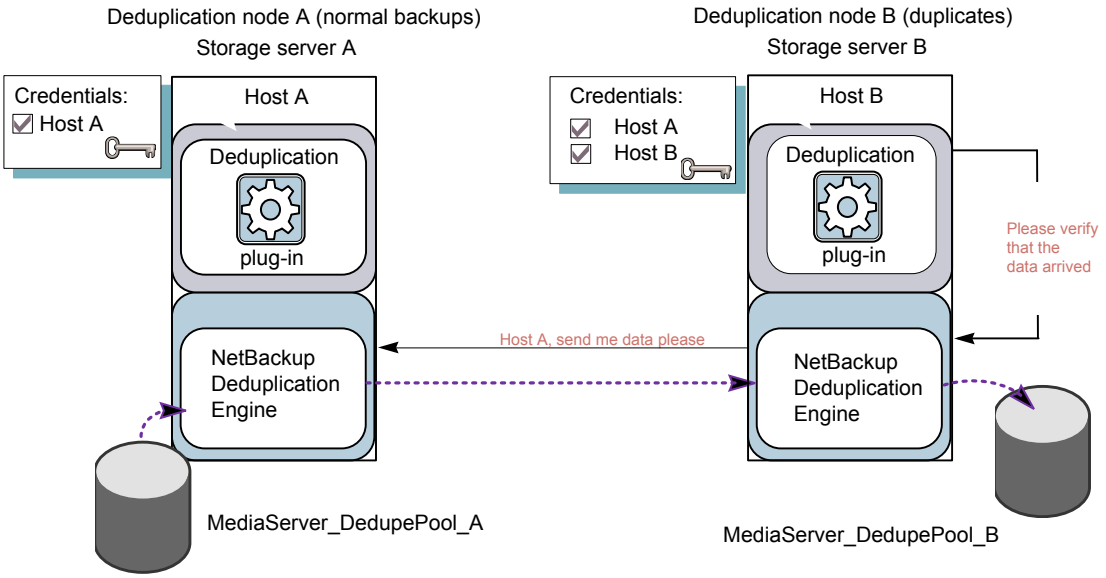
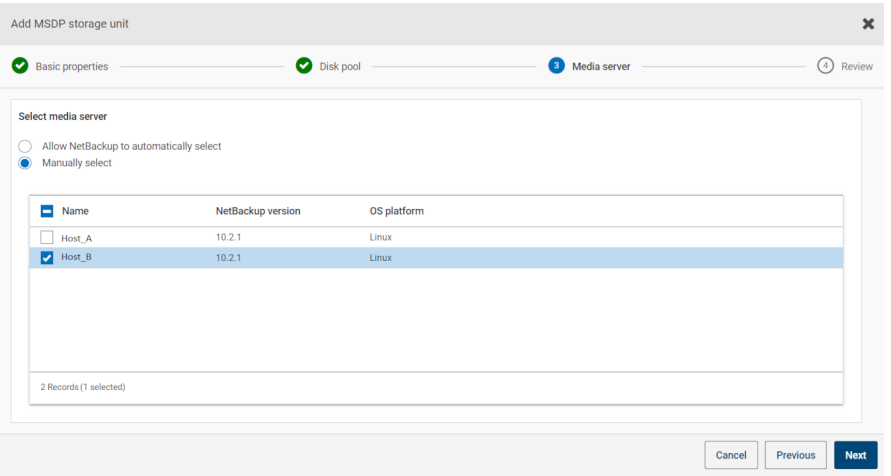


Figure 4-5 shows the storage unit settings for the duplication destination. They are similar to the push example except host B is selected. Host B is the common server, so it must be selected in the storage unit.

Figure 4-5 Pull duplication storage unit settings



If you use node B for backups also, select host B and not host A in the storage unit for the node B backups. If you select host A, it becomes a load balancing server for the node B deduplication pool.

Configuring MSDP optimized duplication within the same NetBackup domain

You can configure optimized duplication from a **Media Server Deduplication Pool** to other deduplication storage within the same NetBackup domain.

Table 4-24 How to configure optimized duplication of deduplicated data

Step	Action	Description
Step 1	Review optimized duplication	See “About MSDP optimized duplication within the same domain” on page 122.

Table 4-24 How to configure optimized duplication of deduplicated data *(continued)*

Step	Action	Description
Step 2	Configure the storage servers	See "Configuring a storage server for a Media Server Deduplication Pool" on page 93. One server must be common between the source storage and the destination storage. Which you choose depends on whether you want a push or a pull configuration. See "About the media servers for MSDP optimized duplication within the same domain" on page 124. For a push configuration, configure the common server as a load balancing server for the storage server for your normal backups. For a pull configuration, configure the common server as a load balancing server for the storage server for the copies at your remote site. Alternatively, you can add a server later to either environment. (A server becomes a load balancing server when you select it in the storage unit for the deduplication pool.)
Step 3	Configure the deduplication pools	If you did not configure the deduplication pools when you configured the storage servers, use the **Disk Pool tab** to configure them. See "Configuring a disk pool for deduplication" on page 98.
Step 4	Configure the storage unit for backups	In the storage unit for your backups, do the following: **1** For the **Disk type**, select **PureDisk**. **2** For the **Disk pool**, select your **Media Server Deduplication Pool**. If you use a pull configuration, do not select the common media server in the backup storage unit. If you do, NetBackup uses it to deduplicate backup data. (That is, unless you want to use it for a load balancing server for the source deduplication node.)

Table 4-24 How to configure optimized duplication of deduplicated data *(continued)*

Step	Action	Description
Step 5	Configure the storage unit for duplication	Cohesity recommends that you configure a storage unit specifically to be the target for the optimized duplication. Configure the storage unit in the deduplication node that performs your normal backups. Do not configure it in the node that contains the copies. In the storage unit that is the destination for your duplicated images, do the following: **1** For the **Disk type**, select **PureDisk**. **2** For the **Disk pool**, the destination can be a **Media Server Deduplication Pool**. Also select **Only use the following media servers**. Then, select the media server or media servers that are common to both the source storage server and the destination storage server. If you select more than one, NetBackup assigns the duplication job to the least busy media server. If you select only a media server (or servers) that is not common, the optimized duplication job fails.
Step 6	Configure optimized duplication bandwidth	Optionally, you can configure the bandwidth for replication. See "About configuring MSDP optimized duplication and replication bandwidth" on page 160.
Step 7	Configure optimized duplication behaviors	Optionally, you can configure the optimized duplication behavior. See "Configuring NetBackup optimized duplication or replication behavior" on page 133. See "About configuring MSDP optimized duplication and replication bandwidth" on page 160.

Table 4-24 How to configure optimized duplication of deduplicated data *(continued)*

Step	Action	Description
Step 8	Configure a storage lifecycle policy for the duplication	Configure a storage lifecycle policy only if you want to use one to duplicate images. The storage lifecycle policy manages both the backup jobs and the duplication jobs. Configure the lifecycle policy in the deduplication environment that performs your normal backups. Do not configure it in the environment that contains the copies. When you configure the storage lifecycle policy, do the following: ■ The first operation must be a **Backup**. For the **Storage** for the **Backup** operation, select the storage unit that is the target of your backups. That storage unit can use a **Media Server Deduplication Pool**. These backups are the primary backup copies; they are the source images for the duplication operation. ■ For the second, child **Operation**, select **Duplication**. Then, select the storage unit for the destination deduplication pool. That pool may can be a **Media Server Deduplication Pool**. See "About storage lifecycle policies" on page 162. See "Creating a storage lifecycle policy" on page 165.
Step 9	Configure a backup policy	Configure a policy to back up your clients. Configure the backup policy in the deduplication environment that performs your normal backups. Do not configure it in the environment that contains the copies. ■ If you use a storage lifecycle policy to manage the backup job and the duplication job: Select that storage lifecycle policy in the **Policy storage** field of the Policy **Attributes** tab. ■ If you do not use a storage lifecycle policy to manage the backup job and the duplication job: Select the storage unit that contains your normal backups. These backups are the primary backup copies. See "About MSDP backup policy configuration" on page 169.

Table 4-24 How to configure optimized duplication of deduplicated data *(continued)*

Step	Action	Description
Step 10	Configure NetBackup Vault for the duplication	Configure Vault duplication only if you use NetBackup Vault to duplicate the images. Configure Vault in the deduplication environment that performs your normal backups. Do not configure it in the environment that contains the copies. For Vault, you must configure a Vault profile and a Vault policy. ■ Configure a Vault profile. ■ On the Vault **Profile** dialog box **Choose Backups** tab, choose the backup images in the source **Media Server Deduplication Pool**. ■ On the **Profile** dialog box **Duplication** tab, select the destination storage unit in the **Destination Storage unit** field. ■ Configure a Vault policy to schedule the duplication jobs. A Vault policy is a NetBackup policy that is configured to run Vault jobs.
Step 11	Duplicate by using the `bpduplicate` command	Use the NetBackup `bpduplicate` command only if you want to duplicate images manually. Duplicate from a **Media Server Deduplication Pool** or a **PureDisk Deduplication Pool** to another **Media Server Deduplication Pool** in the same domain. See the *NetBackup Commands Reference Guide*: https://support.cohesity.com/s/article/article-100040135

Configuring NetBackup optimized duplication or replication behavior

You can configure some optimized duplication and replication behaviors for NetBackup. The behaviors depend on how NetBackup duplicates the images, as described in the following table.

Table 4-25 Optimized duplication behavior

Behavior	Description
Duplication by using NetBackup Vault or the `bpduplicate` command	If you use NetBackup Vault or the `bpduplicate` command for duplication, you can configure the following behaviors: ■ Number of optimized duplication attempts. You can change the number of times NetBackup retries an optimized duplication job before it fails the jobs. See "To configure the number of duplication attempts" on page 134. ■ Optimized duplication failover. By default, if an optimized duplication job fails, NetBackup does not run the job again. You can configure NetBackup to use normal duplication if an optimized duplication job fails. See "To configure optimized duplication failover" on page 135.
Duplication or replication by using a storage lifecycle policy	If a storage lifecycle policy optimized duplication or replication job fails, NetBackup waits 2 hours and retries the job again. NetBackup repeats the retry behavior until the job succeeds or the source backup image expire. You can change the number of hours for the wait period. See "To configure the storage lifecycle policy wait period" on page 135.

If you use a storage lifecycle policy for duplication, do not configure optimized duplication behavior for NetBackup Vault or the `bpduplicate` command, and vice versa. NetBackup behavior may not be predictable.

Caution: These settings affect all optimized duplication jobs; they are not limited to a specific NetBackup storage option.

To configure the number of duplication attempts

◆ On the primary server, create a file named `OPT_DUP_BUSY_RETRY_LIMIT`. Add an integer to the file that specifies the number of times to retry the job before NetBackup fails the job.

The file must reside on the primary server in the following directory (depending on the operating system):

- UNIX: `/usr/openv/netbackup/db/config`

- Windows: *install_path*\NetBackup\db\config.

To configure optimized duplication failover

- On the primary server, add the following configuration option:

  RESUME_ORIG_DUP_ON_OPT_DUP_FAIL = TRUE

  See "Setting NetBackup configuration options by using the command line" on page 135.

  Alternatively on UNIX systems, you can add the entry to the bp.conf file on the NetBackup primary server.

To configure the storage lifecycle policy wait period

1. In NetBackup web UI, select **Host > Host properties**.
2. Select the host to edit and click **Edit primary server**.
3. Select **SLP settings**.
4. Change the **Extended image retry interval** to the new value.
5. Click **Save**.

Setting NetBackup configuration options by using the command line

Cohesity recommends to use the NetBackup web UI to configure the host properties.

However, some properties cannot be set by using the **NetBackup web UI**. You can set those properties by using the following NetBackup commands:

For a NetBackup server: bpsetconfig

For a NetBackup client: nbsetconfig

Configuration options are key and value pairs, as shown in the following examples:

- CLIENT_READ_TIMEOUT = 300
- LOCAL_CACHE = NO
- RESUME_ORIG_DUP_ON_OPT_DUP_FAIL = TRUE
- SERVER = server1.example.com

You can specify some options multiple times, such as the SERVER option.

To set configuration options by using the command line

1. In a command window or shell window on the host on which you want to set the property, invoke the appropriate command. The command depends on the operating system and the NetBackup host type (client or server), as follows:

   UNIX — On a NetBackup client:

   ```
   /usr/openv/netbackup/bin/nbsetconfig
   ```

   On a NetBackup server:

   ```
   /usr/openv/netbackup/bin/admincmd/bpsetconfig
   ```

   Windows — On a NetBackup client:

   ```
   install_path\NetBackup\bin\nbsetconfig.exe
   ```

   On a NetBackup server:

   ```
   install_path\NetBackup\bin\admincmd\bpsetconfig.exe
   ```

2. At the command prompt, enter the key and the value pairs of the configuration options that you want to set, one pair per line.

   You can change existing key and value pairs.

   You can add key and value pairs.

   Ensure that you understand the values that are allowed and the format of any new options that you add.

3. To save the configuration changes, type the following, depending on the operating system:

   Windows: `Ctrl + Z Enter`

   UNIX: `Ctrl + D Enter`

About MSDP replication to a different domain

NetBackup supports replication to storage in a different domain. NetBackup Auto Image Replication is the method used to replicate backup images. (Backup image replication is not the same as snapshot replication, which may occur in the same domain.) You can replicate from one source to one or more destinations.

Table 4-26 describes the MSDP replication source and targets that NetBackup supports.

Table 4-26 NetBackup media server deduplication replication targets

Source storage	Target storage
Media Server Deduplication Pool	A **Media Server Deduplication Pool**, which can be hosted on the following systems: ■ A NetBackup media server. ■ A NetBackup 5200 series appliance or NetBackup 5300 series appliance.

Auto Image Replication does not support replicating from a storage unit group. That is, the source copy cannot be in a storage unit group.

If a replication job fails, NetBackup retries the replication until it succeeds or the source images expire. You can change the retry interval behavior.

See "Configuring NetBackup optimized duplication or replication behavior" on page 133.

If a job fails after it replicates some of the images, NetBackup does not run a separate image cleanup job for the partially replicated images. The next time the replication runs, that job cleans up the image fragments before it begins to replicate the images.

You can use a separate network for the duplication traffic.

See "About a separate network path for MSDP duplication and replication" on page 120.

See "Configuring MSDP replication to a different NetBackup domain" on page 137.

See "About MSDP optimized duplication and replication" on page 45.

Configuring MSDP replication to a different NetBackup domain

Table 4-27 describes the tasks that are required to replicate backup images from one **Media Server Deduplication Pool** to another in a different NetBackup domain.

Optionally, you can use a separate network for the optimized duplication traffic.

See "About a separate network path for MSDP duplication and replication" on page 120.

Table 4-27 NetBackup MSDP replication configuration tasks

Step	Task	Procedure
Step 1	Learn about MSDP replication	See "About MSDP replication to a different domain" on page 136. See "About NetBackup Auto Image Replication" on page 139.
Step 2	Determine if you need to configure a trust relationship with the target NetBackup domain	A trust relationship is optional. See "About trusted primary servers for Auto Image Replication" on page 146.
Step 3	Add the remote storage server as a replication target	See "Configuring a target for MSDP replication to a remote domain" on page 156. See "Viewing the replication topology for Auto Image Replication" on page 144.
Step 4	Configure a storage lifecycle policy	The following are the options when you configure the SLP operations: ■ If you configured a trust relationship with the target domains, you can specify one of the following options: ■ **All replication target storage servers (across different NetBackup domains)** NetBackup automatically creates an import SLP in the target domain when the replication job runs. ■ **A specific Master Server**. If you choose this option, you then select **Target master server** and **Target import SLP**. You must create an import SLP in the target domain before you configure an SLP in the source domain. ■ If you did *not* configure a trust relationship with the target domains, **All replication target storage servers (across different NetBackup domains)** is selected by default. You cannot choose a specific target storage server. NetBackup automatically creates an import SLP in the target domain when the replication job runs. See "About storage lifecycle policies" on page 162. See "About the storage lifecycle policies required for Auto Image Replication" on page 163. See "Creating a storage lifecycle policy" on page 165.
	Configure replication bandwidth	Optionally, you can configure the bandwidth for replication. See "About configuring MSDP optimized duplication and replication bandwidth" on page 160.

About NetBackup Auto Image Replication

The backups that are generated in one NetBackup domain can be replicated to storage in one or more target NetBackup domains. This process is referred to as Auto Image Replication.

The ability to replicate backups to storage in other NetBackup domains, often across various geographical sites, helps facilitate the following disaster recovery needs:

- One-to-one model
  A single production data center can back up to a disaster recovery site.
- One-to-many model
  A single production data center can back up to multiple disaster recovery sites.
  See "One-to-many Auto Image Replication model" on page 140.
- Many-to-one model
  Remote offices in multiple domains can back up to a storage device in a single domain.
- Many-to-many model
  Remote data centers in multiple domains can back up multiple disaster recovery sites.

NetBackup supports Auto Image Replication from a disk volume in a **Media Server Deduplication Pool** in one NetBackup domain to a disk volume in a **Media Server Deduplication Pool** in another domain.

Notes about Auto Image Replication

- Auto Image Replication does not support synthetic backups or optimized synthetic backups.
- Auto Image Replication does not support spanning volumes in a disk pool. NetBackup fails backup jobs to the disk pools that span volumes if the backup job is in a storage lifecycle policy that also contains a replication operation.
- Auto Image Replication does not support replicating from a storage unit group. That is, the source copy cannot be in a storage unit group.
- The ability to perform Auto Image Replication between different versions of NetBackup does not overrule the basic image compatibility rules. For example, a database backup that was taken in one NetBackup domain can be replicated to a NetBackup domain of an earlier version. However, the older server may not be able to successfully restore from the newer image.
  For information about version compatibility and interoperability, see the NetBackup Enterprise Server and Server Software Compatibility List at the following URL:
  http://www.netbackup.com/compatibility

- Synchronize the clocks of the primary servers in the source and the target domains so that the primary server in the target domain can import the images as soon as they are ready. The primary server in the target domain cannot import an image until the image creation time is reached. Time zone differences are not a factor because the images use Coordinated Universal Time (UTC).

Process Overview

Table 4-28 is an overview of the process, generally describing the events in the originating and target domains.

NetBackup uses storage lifecycle policies in the source domain and the target domain to manage the Auto Image Replication operations.

See "About the storage lifecycle policies required for Auto Image Replication" on page 163.

Table 4-28 Auto Image Replication process overview

Event	Domain in which event occurs	Event description
1	Originating primary server (Domain 1)	Clients are backed up according to a backup policy that indicates a storage lifecycle policy as the **Policy storage** selection. The SLP must include at least one **Replication** operation to similar storage in the target domain.
2	Target primary server (Domain 2)	The storage server in the target domain recognizes that a replication event has occurred. It notifies the NetBackup primary server in the target domain.
3	Target primary server (Domain 2)	NetBackup imports the image immediately, based on an SLP that contains an import operation. NetBackup can import the image quickly because the metadata is replicated as part of the image. (This import process is not the same as the import process available in the **Catalog** utility.)
4	Target primary server (Domain 2)	After the image is imported into the target domain, NetBackup continues to manage the copies in that domain. Depending on the configuration, the media server in Domain 2 can replicate the images to a media server in Domain 3.

One-to-many Auto Image Replication model

In this configuration, all copies are made in parallel. The copies are made within the context of one NetBackup job and simultaneously within the originating storage server context. If one target storage server fails, the entire job fails and is retried later.

All copies have the same **Target Retention**. To achieve different **Target Retention** settings in each target primary server domain, either create multiple source copies or cascade duplication to target primary servers.

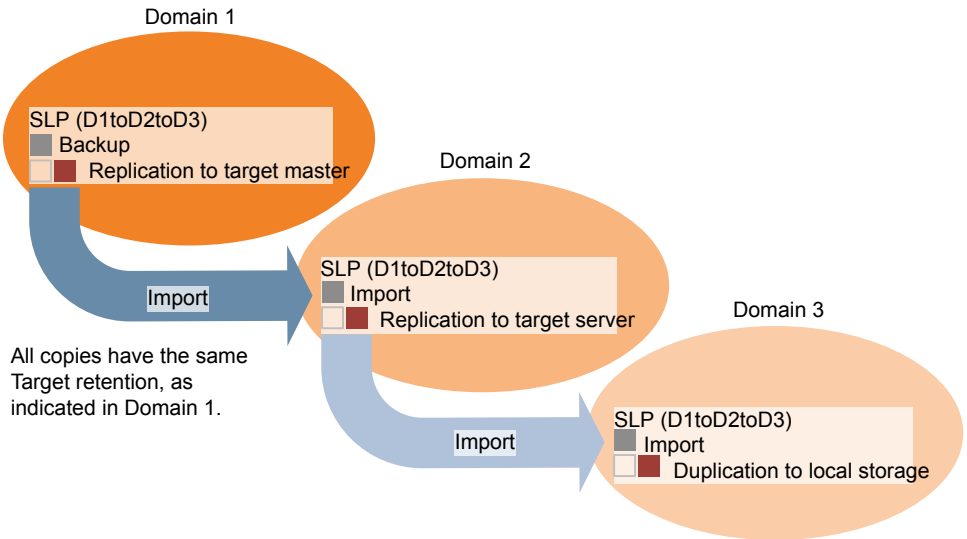
Cascading Auto Image Replication model

Replications can be cascaded from the originating domain to multiple domains. Storage lifecycle policies are set up in each domain to anticipate the originating image, import it and then replicate it to the next target primary.

Figure 4-6 represents the following cascading configuration across three domains.

- The image is created in Domain 1, and then replicated to the target Domain 2.
- The image is imported in Domain 2, and then replicated to a target Domain 3.
- The image is then imported into Domain 3.

Figure 4-6 Cascading Auto Image Replication



In the cascading model, the originating primary server for Domain 2 and Domain 3 is the primary server in Domain 1.

Note: When the image is replicated in Domain 3, the replication notification event indicates that the primary server in Domain 2 is the originating primary server. However, after the image is imported successfully into Domain 3, NetBackup correctly indicates that the originating primary server is in Domain 1.

The cascading model presents a special case for the Import SLP that replicates the imported copy to a target primary. (This primary server that is neither the first nor the last in the string of target primary servers.)

The Import SLP must include at least one operation that uses a **Fixed** retention type and at least one operation that uses a **Target Retention** type. So that the Import SLP can satisfy these requirements, the import operation must use a **Target Retention**.

Table 4-29 shows the difference in the import operation setup.

Table 4-29 Import operation difference in an SLP configured to replicate the imported copy

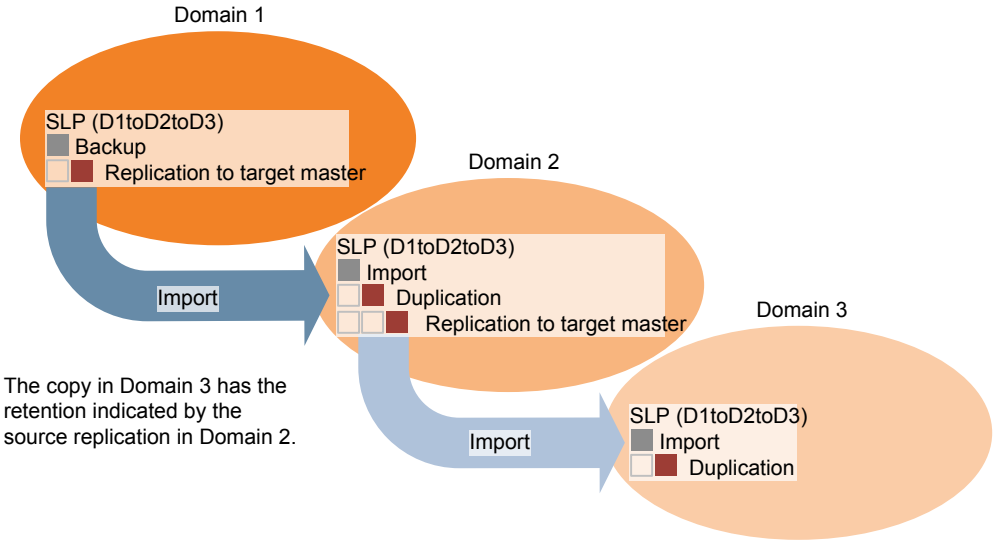
Import operation criteria	Import operation in a cascading model
The first operation must be an import operation.	Same; no difference.
A replication to target primary must use a **Fixed** retention type	Same; no difference.
At least one operation must use the **Target retention**.	Here is the difference: To meet the criteria, the import operation must use **Target retention**.

The target retention is embedded in the source image.

In the cascading model that is represented in Figure 4-6, all copies have the same **Target Retention**—the **Target Retention** indicated in Domain 1.

For the copy in Domain 3 to have a different target retention, add an intermediary replication operation to the Domain 2 storage lifecycle policy. The intermediary replication operation acts as the source for the replication to target primary. Since the target retention is embedded in the source image, the copy in Domain 3 honors the retention level that is set for the intermediary replication operation.

Figure 4-7 Cascading replications to target primary servers, with various target retentions



About the domain relationship for replication

For a **Media Server Deduplication Pool** target: the relationship between the originating domain and the target domain or domains is established in the originating domain. Specifically, by configuring the target storage server in the **Replication** tab of the **Change Storage Server** dialog box of the source storage server.

See "Configuring a target for MSDP replication to a remote domain" on page 156.

Before you configure the replication relationship, you can add the target primary server as a trusted host.

See "About trusted primary servers for Auto Image Replication" on page 146.

Caution: Choose the target storage server carefully. A target storage server must not also be a storage server for the originating domain.

About the replication topology for Auto Image Replication

For Auto Image Replication, the disk volumes have the properties that define the replication relationships between the volumes. The knowledge of the volume properties is considered the replication topology. The following are the replication properties that a volume can have:

Source	A source volume contains the backups of your clients. The volume is the source for the images that are replicated to a remote NetBackup domain. Each source volume in an originating domain has one or more replication partner target volumes in a target domain.
Target	A target volume in the remote domain is the replication partner of a source volume in the originating domain.
None	The volume does not have a replication attribute.

NetBackup exposes the storage for a **Media Server Deduplication Pool** as a single volume. Therefore, there is always a one-to-one volume relationship for MSDP.

You configure the replication relationships in the source domain.

See "Configuring a target for MSDP replication to a remote domain" on page 156.

NetBackup discovers the replication topology when you configure the replication relationships. NetBackup discovers topology changes when you use the **Refresh** option in **Storage > Disk storage > Disk pools**.

See "Changing a Media Server Deduplication Pool properties" on page 544.

NetBackup includes a command that can help you understand your replication topology. Use the command in the following situations:

- After you configure the replication targets.
- After you configure the storage server and before you configure disk pools.
- After changes to the volumes that comprise the storage.

See "Viewing the replication topology for Auto Image Replication" on page 144.

Viewing the replication topology for Auto Image Replication

A volume that is a source of replication must have at least one replication partner that is the target of the replication. NetBackup lets you view the replication topology of the storage.

See "About the replication topology for Auto Image Replication" on page 143.

To view the replication topology for Auto Image Replication

- Run the `bpstsinfo` command, specifying the storage server name and the server type. The following is the command syntax:
  - Windows: *`install_path`*`\NetBackup\bin\admincmd\bpstsinfo -lsuinfo -storage_server` *`host_name`* `-stype` *`server_type`*

- UNIX: `/usr/openv/netbackup/bin/admincmd/bpstsinfo -lsuinfo -storage_server` *`host_name`* `-stype` *`server_type`*

The following are the options and arguments for the command:

`-storage_server` ***host_name***	The name of the target storage server.
`-stype` **PureDisk**	Use `PureDisk` for a **Media Server Deduplication Pool**,.

Save the output to a file so that you can compare the current topology with the previous topology to determine what has changed.

See "Sample volume properties output for MSDP replication" on page 145.

Sample volume properties output for MSDP replication

The following two examples show output from the `bpstsinfo -lsuinfo` command for two NetBackup deduplication storage servers. The first example is the output from the source disk pool in the originating domain. The second example is from the target disk pool in the remote primary server domain.

The two examples show the following:

- All of the storage in a deduplication disk pool is exposed as one volume: `PureDiskVolume`.
- The `PureDiskVolume` of the deduplication storage server `bit1.datacenter.example.com` is the source for the replication operation.
- The `PureDiskVolume` of the deduplication storage server `target_host.dr-site.example.com` is the target of the replication operation.

```
> bpstsinfo -lsuinfo -storage_server bit1.datacenter.example.com -stype PureDisk
LSU Info:
        Server Name: PureDisk:bit1.datacenter.example.com
        LSU Name: PureDiskVolume
        Allocation : STS_LSU_AT_STATIC
        Storage: STS_LSU_ST_NONE
        Description: PureDisk storage unit (/bit1.datacenter.example.com#1/2)
        Configuration:
        Media: (STS_LSUF_DISK | STS_LSUF_ACTIVE | STS_LSUF_STORAGE_NOT_FREED |
         STS_LSUF_REP_ENABLED | STS_LSUF_REP_SOURCE)
        Save As : (STS_SA_CLEARF | STS_SA_IMAGE | STS_SA_OPAQUEF)
        Replication Sources: 0 ( )
        Replication Targets: 1 ( PureDisk:target_host.dr-site.example.com:PureDiskVolume )
        Maximum Transfer: 2147483647
        Block Size: 512
```

```
        Allocation Size: 0
        Size: 74645270666
        Physical Size: 77304328192
        Bytes Used: 138
        Physical Bytes Used: 2659057664
        Resident Images: 0

> bpstsinfo -lsuinfo -storage_server target_host.dr-site.example.com -stype PureDisk
LSU Info:
        Server Name: PureDisk:target_host.dr-site.example.com
        LSU Name: PureDiskVolume
        Allocation : STS_LSU_AT_STATIC
        Storage: STS_LSU_ST_NONE
        Description: PureDisk storage unit (/target_host.dr-site.example.com#1/2)
        Configuration:
        Media: (STS_LSUF_DISK | STS_LSUF_ACTIVE | STS_LSUF_STORAGE_NOT_FREED |
         STS_LSUF_REP_ENABLED | STS_LSUF_REP_TARGET)
        Save As : (STS_SA_CLEARF | STS_SA_IMAGE | STS_SA_OPAQUEF)
        Replication Sources: 1 ( PureDisk:bit1:PureDiskVolume )
        Replication Targets: 0 ( )
        Maximum Transfer: 2147483647
        Block Size: 512
        Allocation Size: 0
        Size: 79808086154
        Physical Size: 98944983040
        Bytes Used: 138
        Physical Bytes Used: 19136897024
        Resident Images: 0
```

About trusted primary servers for Auto Image Replication

NetBackup provides the ability to establish a trust relationship between replication domains. A trust relationship is optional for the Media Server Deduplication Pool as a target storage. Before you configure a storage server as a target storage, establish a trust relationship between the source A.I.R. and the target A.I.R operations.

The following items describe how a trust relationship affects Auto Image Replication:

No trust relationship	NetBackup replicates to all defined target storage servers. You cannot select a specific host or hosts as a target.

Trust relationship	You can select a subset of your trusted domains as a target for replication. NetBackup then replicates to the specified domains only rather than to all configured replication targets. This type of Auto Image Replication is known as targeted A.I.R.

About adding a trusted primary server using NetBackup CA-signed certificate

With targeted A.I.R., when trust is established between the source and the remote target server, you need to establish trust in both the domains.

1. In the source primary server, add the target primary server as a trusted server.
2. In the target primary server, add the source primary server as a trusted server.

Note: The **NetBackup web UI** does not support adding a trusted primary server using an external CA-signed certificate.

See “Add a trusted primary server” on page 151.

See “About the certificate to use to add a trusted primary server” on page 150.

The following diagram illustrates the different tasks for adding trusted primary servers when NetBackup CA-signed certificate (or host ID-based certificate) is used to establish trust between the source and the target primary servers.

Figure 4-8 Tasks to establish a trust relationship between primary servers for targeted A.I.R. using NetBackup CA-signed certificate

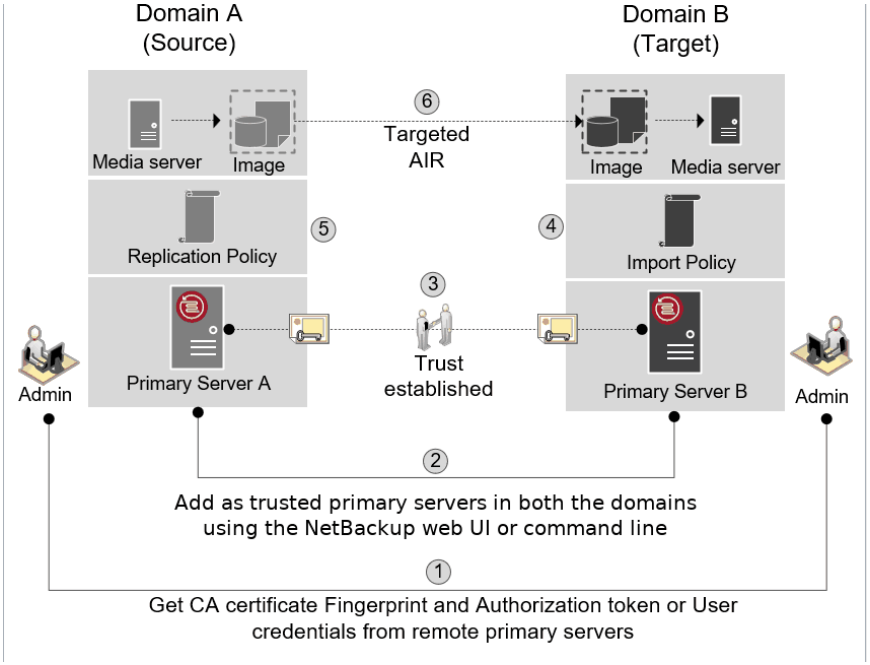


Table 4-30 Tasks to establish a trust relationship between primary servers for targeted A.I.R.

Step	Task	Procedure
Step 1	Administrators of both the source and the target primary servers must obtain each other's CA certificate fingerprint and authorization tokens or the user credentials. This activity must be performed offline. **Note:** It is recommended to use an authentication token to connect to the remote primary server. An authentication token provides restricted access and allows secure communication between both the hosts. The use of user credentials (user name and password) may present a possible security breach.	To obtain the authorization tokens, use the `bpnbat` command to log on and `nbcertcmd` to get the authorization tokens. To obtain the SHA1 fingerprint of root certificate, use the `nbcertcmd -displayCACertDetail` command. To perform this task, see the NetBackup Commands Reference Guide. **Note:** When you run the commands, keep the target as the remote server.

Table 4-30 Tasks to establish a trust relationship between primary servers for targeted A.I.R. *(continued)*

Step	Task	Procedure
Step 2	Establish trust between the source and the target domains. ■ On the source primary server, add the target primary server as trusted server. ■ On the target primary server, add the source primary server as trusted server.	To perform this task in the NetBackup web UI, see the following topic: See "Add a trusted primary server" on page 151. To perform this task using the `nbseccmd`, see the NetBackup Commands Reference Guide.
Step 3	After you have added the source and target trusted servers, they have each other's host ID-based certificates. The certificates are used during each communication. Primary Server A has a certificate that Primary Server B issued and vice versa. Before communication can occur, Primary Server A presents the certificate that Primary Server B issued and vice versa. The communication between the source and the target primary servers is now secured.	To understand the use of host ID-based certificates, see the NetBackup Security and Encryption Guide.
Step 3.1	Configure the source media server to get the security certificates and the host ID certificates from the target primary server.	See "Configuring NetBackup CA and NetBackup host ID-based certificate for secure communication between the source and the target MSDP storage servers" on page 154. See "Configuring a NetBackup Deduplication Engine user with limited permissions for Auto Image Replication" on page 160.
Step 4	Create an import storage lifecycle policy in the target domain. **Note:** The import storage lifecycle policy name should contain less than or equal to 112 characters.	See "About storage lifecycle policies" on page 162.
Step 5	On the source MSDP server, add the credentials of the target storage server.	See "Configuring a target for MSDP replication to a remote domain" on page 156.
Step 5.1	Create a replication storage lifecycle policy in the source domain using the specific target primary server and storage lifecycle policy. The backups that are generated in one NetBackup domain can be replicated to storage in one or more target NetBackup domains.	See "About storage lifecycle policies" on page 162.

Table 4-30 Tasks to establish a trust relationship between primary servers for targeted A.I.R. *(continued)*

Step	Task	Procedure
Step 6	The backups that are generated in one NetBackup domain can be replicated to storage in one or more target NetBackup domains. This process is referred to as Auto Image Replication.	See "About NetBackup Auto Image Replication" on page 139.

About the certificate to use to add a trusted primary server

A source or a target primary server may use NetBackup CA-signed certificates (host ID-based certificates) or external CA-signed certificates.

For more information on NetBackup host ID-based certificates and external CA support, refer to the NetBackup Security and Encryption Guide.

To establish trust between source and target primary servers, NetBackup verifies the following:

Can the source primary server establish trust using an external CA-signed certificate?	If the external CA configuration options - `ECA_CERT_PATH`, `ECA_PRIVATE_KEY_PATH`, and `ECA_TRUST_STORE_PATH` - are defined in the NetBackup configuration file of the source primary server, it can establish the trust using an external certificate. In the case of the Windows certificate trust store, only the option `ECA_CERT_PATH` is defined.
Which certificate authorities (CA) does the target primary server support?	The target primary server may support external CA, NetBackup CA, or both.

The following table lists the CA support scenarios and the certificate to use to establish trust between the source and the target primary servers. The instructions assume that you use the NetBackup web UI for the configuration.

Table 4-31 Certificate to use for the trust setup

Can the primary server use an external certificate?	Which CA does the target primary server use?	Certificate to use for the trust setup
Yes The source primary server can use the NetBackup CA and an external CA for communication with a remote primary server.	External CA	External CA
	NetBackup CA	NetBackup CA
	External CA and NetBackup CA	NetBackup prompts to select the CA that you want to use for trust setup.
No The source primary server can only use the NetBackup CA for communication with a remote primary server.	External CA	No trust is established.
	NetBackup CA	NetBackup CA
	External CA and NetBackup CA	NetBackup CA

Add a trusted primary server

Replication operations require that a trust relationship exists between the NetBackup servers in the different domains. You can create a trust relationship between the primary servers that both use the NetBackup CA or that both use an external CA.

Before you begin, review the following information:

- Ensure that you have the RBAC System Administrator role or a role with similar permissions. Or, for appliances with software versions 3.1 and later you must have permissions for the NetBackup CLI user.
- For a remote Windows primary server, the user's domain may not be the same as that of the authentication service. In this case you must add the domain with LDAP using the `vssat addldapdomain` command.
- For a NetBackup CA-signed certificate, the recommended method to authenticate the server is the option **Specify authentication token of the trusted primary server**.
- If you use the option **Specify credentials of the trusted primary server**, that method may present a possible security breach. Only an authentication token can provide restricted access and allow secure communication between both the hosts. To establish trust with a 3.1 NetBackup primary appliance, use the NetBackup CLI credentials.

To add a trusted primary server

1. Open the NetBackup web UI.
2. Identify the NetBackup versions that are installed and the certificate types that are used on the source and the target servers.

   Both servers must use the same certificate type.
3. For the servers that use the NetBackup certificate authority (CA), obtain an authorization token for the remote server.
4. For the servers that use the NetBackup certificate authority (CA), obtain the fingerprint for each server.
5. At the top right, select **Settings > Global security**.
6. Select the **Trusted primary servers** tab.
7. Select the **Add** button.
8. Enter the fully-qualified host name of the remote primary server and select **Validate Certificate Authority**.
9. Follow the prompts in the UI.
10. Repeat these steps on the remote primary server.

More information

For more information on using an external CA with NetBackup, see the NetBackup Security and Encryption Guide.

Remove a trusted primary server

You can remove a trusted primary server, which removes the trust relationship between primary servers. Note the following implications:

- Any replication operations fail that require the trust relationship.
- A remote primary server is not included in any usage reporting after you remove the trust relationship.

To remove a trusted primary server, you must perform the following procedure on both the source and the target server.

To remove a trusted primary server

1. Open the NetBackup web UI.
2. Ensure that all replication jobs to the target primary server are complete.

3 Delete all storage lifecycle policies (SLPs) that use the trusted primary as a destination. Before deleting an SLP, ensure that there are no backup policies or protection plans that use the SLP for storage.

4 At the top right, select **Settings > Global security**.

5 Select the **Trusted primary servers** tab.

6 Locate the server that you want to remove.

7 Select **Actions > Remove**.

8 Select **Remove trust**.

Note: If you use multiple NICs, if you established trust using more that one host NIC and if you remove the trust relationship with any one host NIC, the trust with all the other host NICs is broken.

Enable inter-node authentication for a NetBackup clustered primary server

NetBackup requires inter-node authentication among the primary servers in a cluster. For authentication, you must provision an authentication certificate on all of the nodes of the cluster. The certificates are used to establish SSL connections between the NetBackup hosts.

See "Add a trusted primary server" on page 151.

The inter-node authentication allows the following NetBackup functionality:

NetBackup web UI	The NetBackup web UI in primary server clusters requires the NetBackup authentication certificates for correct functionality.
Targeted A.I.R. (Auto Image Replication)	Auto Image Replication in which a primary server is in a cluster requires inter-node authentication among the hosts in that cluster. The NetBackup authentication certificates provide the means to establish the proper trust relationships. Provision the certificates on the cluster hosts before you add the trusted primary server. This requirement applies regardless of whether the clustered primaryed server is the source of the replication operation or the target.

To enable inter-node authentication for a NetBackup clustered primary server

- On the active node of the NetBackup primary server cluster, run the following NetBackup command:

- Windows: *install_path*\NetBackup\bin\admincmd\bpnbaz -setupat
- UNIX: /usr/openv/netbackup/bin/admincmd/bpnbaz -setupat

NetBackup creates the certificates on every node in the primary server cluster.

The following is example output:

```
# bpnbaz -setupat
You will have to restart Netbackup services on this machine after
the command completes successfully.
Do you want to continue(y/n)y
Gathering configuration information.
Please be patient as we wait for 10 sec for the security services
to start their operation.
Generating identity for host 'bit1.remote.example.com'
Setting up security on target host: bit1.remote.example.com
nbatd is successfully configured on Netbackup Primary Server.
Operation completed successfully.
```

Configuring NetBackup CA and NetBackup host ID-based certificate for secure communication between the source and the target MSDP storage servers

MSDP now supports secure communications between two media servers from two different NetBackup domains. The secure communication is set up when you run Auto Image Replication (A.I.R.). The two media servers must use the same CA to do the certificate security check. The source MSDP server uses the CA of the target NetBackup domain and the certificate that is authorized by the target NetBackup domain. You must manually deploy CA and the certificate on the source MSDP server before using Auto Image Replication.

Note: After you upgrade to NetBackup 8.1.2 or later, manually deploy NetBackup CA and the NetBackup host ID-based certificate on the source MSDP server to use the existing Auto Image Replication.

To configure the NetBackup CA and a NetBackup host ID-based certificate, complete the following steps:

1. On the target NetBackup primary server, run the following command to display the NetBackup CA fingerprint:

   - Windows

     install_path\NetBackup\bin\nbcertcmd -displayCACertDetail

- UNIX

  `/usr/openv/netbackup/bin/nbcertcmd -displayCACertDetail`

2. On the source MSDP storage server, run the following command to get the NetBackup CA from target NetBackup primary server:

   - Windows

     install_path`\NetBackup\bin\nbcertcmd -getCACertificate -server` *target_primary_server*

   - UNIX

     `/usr/openv/netbackup/bin/nbcertcmd -getCACertificate -server` *target_primary_server*

   When you accept the CA, ensure that the CA fingerprint is the same as displayed in the previous step.

3. On the source MSDP storage server, run the following command to get a certificate generated by target NetBackup primary server:

   - Windows

     install_path`\NetBackup\bin\nbcertcmd -getCertificate -server` *target_primary_server* `-token` *token_string*

   - UNIX

     `/usr/openv/netbackup/bin/nbcertcmd -getCertificate -server` *target_primary_server* `-token` *token_string*

4. Use either of these two methods to obtain the authorization tokens:

   - NetBackup web UI
     - In NetBackup web UI, select **Security > Tokens**.
     - Click **Add** and fill the required details to create a token.
   - NetBackup commands
     - Use the `bpnbat` command to log on the target NetBackup primary server.
     - Use the `nbcertcmd` command to get the authorization tokens.

     For more information on the commands, refer to the *NetBackup Commands Reference Guide*.

Configuring external CA for secure communication between the source MSDP storage server and the target MSDP storage server

MSDP now supports use of an external CA for secure communication between two media servers that are from two different NetBackup domains. The secure

communication is set up when you run Auto Image Replication (A.I.R.). If the two media servers use different external CAs, then you must exchange the external certificates before you use Auto Image Replication.

To exchange the external certificates, complete the following steps:

1. Copy the root certificate file from the source MSDP storage server to the target MSDP storage server. Combine the certificate files on the target MSDP storage server.
2. Copy the root certificate file from the target MSDP storage server to the source MSDP storage server. Combine the certificate files on the source MSDP storage server.

If the Windows certificate store is used to store the root certificate, add the root certificate to the certificate store. You can use the `certutil` tool to add the root certificate to the certificate store, or just right-click the root certificate file and select **Install Certificate**. When you use the `certutil` tool to install the root certificate, the store name parameter must be **Root**. When you use Windows explorer to install the root certificate, the store location must be **Local Machine** and store name must be **Trusted Root Certification Authorities**.

Configuring a target for MSDP replication to a remote domain

Use the following procedure to configure a target for replication from a **Media Server Deduplication Pool** in an originating domain to a deduplication pool in another target domain. NetBackup supports several deduplication targets.

See "About MSDP replication to a different domain" on page 136.

Configuring the target storage server is only one step in the process of configuring MSDP replication.

See "Configuring MSDP replication to a different NetBackup domain" on page 137.

Note: About clustered primary servers: If you add a trusted primary server for replication operations, you must enable inter-node authentication on all of the nodes in the cluster. Enable the authentication before you begin the following procedure. This requirement applies regardless of whether the clustered primary server is the source of the replication operation or the target.

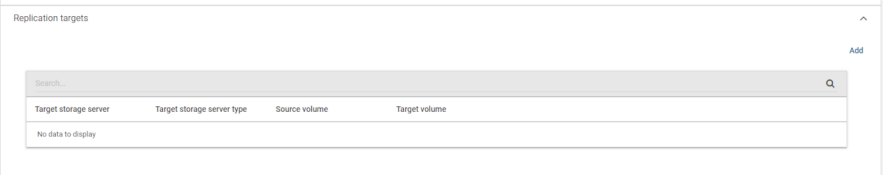
See "About trusted primary servers for Auto Image Replication" on page 146.

See "Enable inter-node authentication for a NetBackup clustered primary server" on page 153.

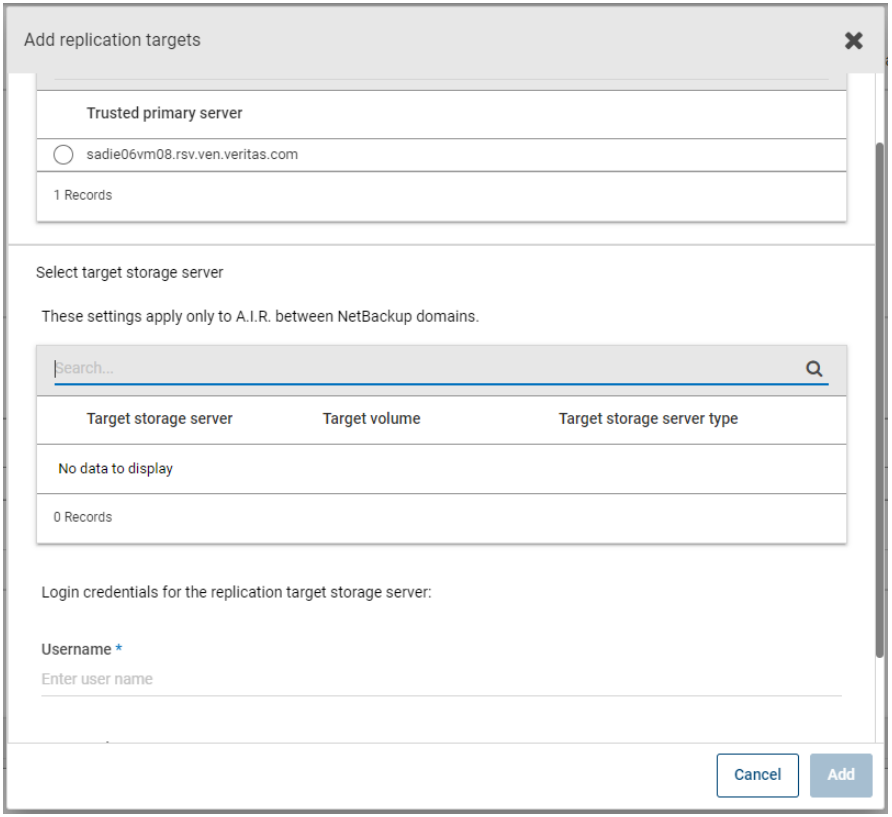
Caution: Choose the target storage server or servers carefully. A target storage server must not also be a storage server for the source domain. Also, a disk volume must not be shared among multiple NetBackup domains.

To configure a Media Server Deduplication Pool as a replication target

1. Open the web UI.
2. On the left, click **Storage > Disk storage**.
3. Click the **Disk pools** tab.
4. Click on the disk pool name.
5. On the **Details** tab locate **Replication targets**. Then click **Add**.



6 Select the trusted primary server.



7 Select the required target storage server.

8 Enter the **Username** and **Password**.

9 Click **Add**.

Configuring a replication target configures the replication properties of the disk volumes in both domains. However, you must refresh the deduplication pools so that NetBackup reads the new volume properties.

See "Changing a Media Server Deduplication Pool properties" on page 544.

Target options for MSDP replication

The following table describes the target options for replication to a NetBackup **Media Server Deduplication Pool**.

Table 4-32 MSDP target replication options

Option	Description
Target master server	All trusted primary servers are in the drop-down list. Select the primary server for the target domain to which you want to replicate backups. To add the primary server of another domain as a trusted primary, select **Add a new Trusted Master Server**. Configuring a trust relationship is required only if you want to choose a specific target for replication.
Target storage server type	If a trusted primary server is configured, the value is **Target storage server name**. If a trusted primary server is not configured, the value is **PureDisk**.
Target storage server name	If a trusted primary server is configured, select the target storage server. If a trusted primary server is *not* configured, enter the name of the target storage server. The drop-down list shows all the storage servers that match the **Target storage server type**.
User name	When you configure a replication target, NetBackup populates the **User name** field with user account of the target storage server, as follows: ■ For an MSDP target, the NetBackup Deduplication Engine user name. For additional security, you can give limited permissions to the deduplication engine user. See "Configuring a NetBackup Deduplication Engine user with limited permissions for Auto Image Replication" on page 160.
Password	Enter the password for the NetBackup Deduplication Engine.

See "Configuring a target for MSDP replication to a remote domain" on page 156.

Configuring a NetBackup Deduplication Engine user with limited permissions for Auto Image Replication

MSDP supports the creation of a user specifically for Auto Image Replication. A user with permissions limited to Auto Image Replication is more secure than a user with administrative permissions.

To configure a NetBackup Deduplication Engine user with limited permissions for Auto Image Replication, complete the following steps:

1. Run the following command on the target MSDP server to add a user for AIR:

   Windows

   ```
   <install_path>/pdde/spauser -a -u <username> -p <password> --role
   air --owner root
   ```

   UNIX

   ```
   /usr/openv/pdde/pdcr/bin/spauser -a -u <username> -p <password>
   --role air --owner root
   ```

2. During configuration of MSDP as a replication target on the source NetBackup primary server, enter the user name and password of the user with limited permissions for A.I.R.

About configuring MSDP optimized duplication and replication bandwidth

Each optimized duplication or Auto Image Replication job is a separate process or stream. The number of duplication or replication jobs that run concurrently determines the number of jobs that contend for bandwidth. You can control how much network bandwidth that optimized duplication and Auto Image Replication jobs consume.

Two different configuration file settings control the bandwidth that is used, as follows:

`bandwidthlimit`

The `bandwidthlimit` parameter in the `agent.cfg` file is the global bandwidth setting. You can use this parameter to limit the bandwidth that all replication jobs use. It applies to jobs in which a **Media Server Deduplication Pool** is the source. Therefore, configure it on the source storage server.

If `bandwidthlimit` is greater than zero, all of the jobs share the bandwidth. That is, the bandwidth for each job is the `bandwidthlimit` divided by the number of jobs.

If `bandwidthlimit=0`, total bandwidth is not limited. However, you can limit the bandwidth that each job uses. See the following `OPTDUP_BANDWIDTH` description.

If you specify bandwidth limits, optimized duplication and replication traffic to any destination is limited.

By default, `bandwidthlimit=0`.

The `agent.cfg` file resides in the following directory:

- UNIX: *`storage_path`*`/etc/puredisk`
- Windows: *`storage_path`*`\etc\puredisk`

The `OPTDUP_BANDWIDTH` parameter in the `pd.conf` file specifies the per job bandwidth.

`OPTDUP_BANDWIDTH` applies only if the `bandwidthlimit` parameter in the `agent.cfg` file is zero.

If `OPTDUP_BANDWIDTH` and `bandwidthlimit` are both `0`, bandwidth per replication job is not limited.

By default, `OPTDUP_BANDWIDTH = 0`.

See "About the MSDP pd.conf configuration file" on page 178.

See "Editing the MSDP pd.conf file" on page 179.

See "MSDP pd.conf file parameters" on page 179.

See "Configuring MSDP optimized duplication within the same NetBackup domain" on page 129.

See "Configuring MSDP replication to a different NetBackup domain" on page 137.

About performance tuning of optimized duplication and replication for large images

When an optimized duplication job or AIR job is initiated to move a large image, tune the `WorkerThreadNumber` and `FragmentGranularity` parameters in the `agent.cfg` file at the source to improve the performance.

The `agent.cfg` file resides at the following directory on the MSDP storage server:

- UNIX: `<storage_path>/etc/puredisk`
- Windows: `<storage_path>\etc\puredisk`

A thread pool is used to improve the concurrency. The `WorkerThreadNumber` parameter defines the number of threads in the thread pool. You can specify values from 0 to 128 for this parameter. The default value is 64. If the value is set to 0, the thread pool is disabled.

The `FragmentGranularity` parameter defines the number of fragments that one thread should handle. You can specify values from 1 to 1048576 for this parameter. The default value is 8.

About performance tuning of optimized duplication and replication for MSDP cloud

When an optimized duplication job or AIR job is initiated from a cloud LSU to a local LSU or another cloud LSU, for high latency network, tune the `MaxPredownloadBatchCount` parameter on the source side to improve the performance.

The `MaxPredownloadBatchCount` parameter in the `agent.cfg` file is the global setting for all cloud LSU. You can tune this parameter to control the concurrency of download from the cloud LSU to improve the performance.

The range of this parameter is from 0 to 100. By default, the value is 20. If the value is set to 0, the concurrent download is disabled.

The `agent.cfg` file resides in the following directory on MSDP storage server:

UNIX: `<storage_path>/etc/puredisk`

About storage lifecycle policies

A storage lifecycle policy (SLP) is a storage plan for a set of backups. An SLP is configured within the **Storage lifecycle policies** utility. It contains instructions in

the form of storage operations, to be applied to the data that is backed up by a backup policy. Operations are added to the SLP that determine how the data is stored, copied, replicated, and retained. NetBackup retries the copies as necessary to ensure that all copies are created.

SLPs offer the opportunity for users to assign a classification to the data at the policy level. A data classification represents a set of backup requirements, which makes it easier to configure backups for data with different requirements. For example, email data and financial data.

SLPs can be set up to provide staged backup behavior. They simplify data management by applying a prescribed behavior to all the backup images that are included in the SLP. This process allows the NetBackup administrator to leverage the advantages of disk-based backups in the near term. It also preserves the advantages of tape-based backups for long-term storage.

The **SLP settings** host properties allow administrators to customize how SLPs are maintained and how SLP jobs run.

Best practice information about SLPs appears in the following document:

https://support.cohesity.com/s/article/article-100009913

For more information, see the NetBackup Web UI Administrator's Guide.

About the storage lifecycle policies required for Auto Image Replication

To replicate images from one NetBackup domain to another NetBackup domain requires two storage lifecycle policies. The following table describes the policies and their requirements:

Table 4-33 SLP requirements for Auto Image Replication

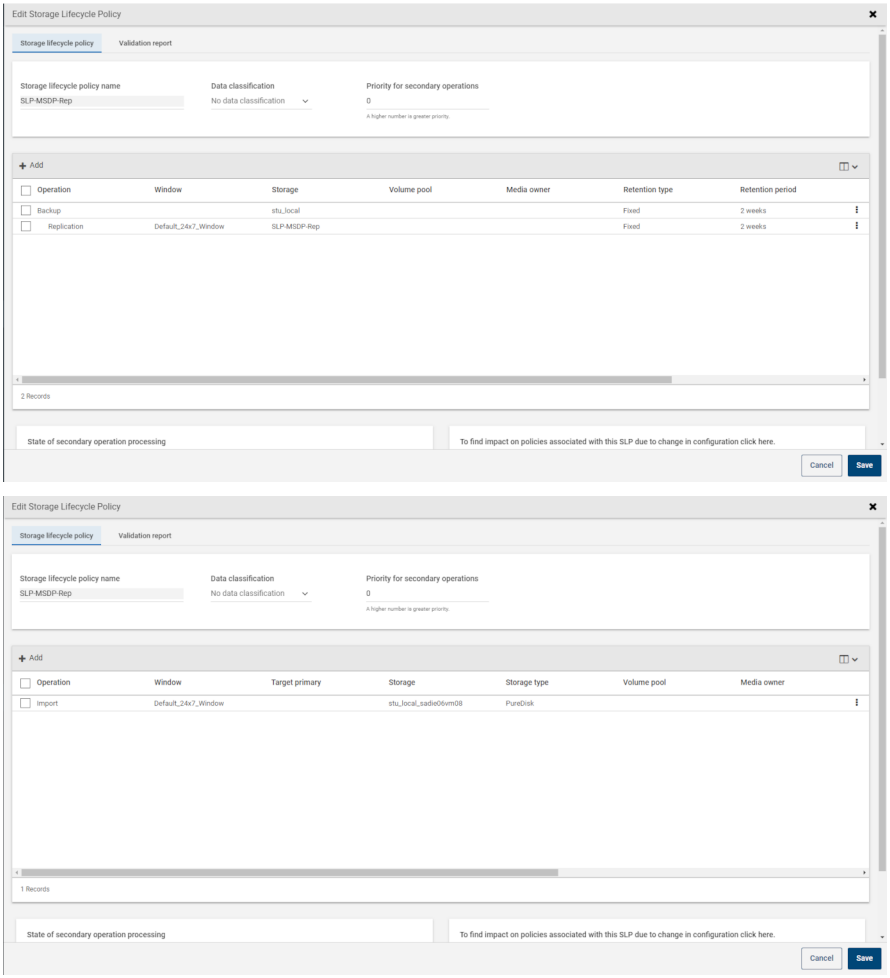
Domain	Storage lifecycle policy requirements
Domain 1 (Source domain)	The Auto Image Replication SLP in the source domain must meet the following criteria: ■ The first operation must be a **Backup** operation to a **Media Server Deduplication Pool**. Indicate the exact storage unit from the drop-down list. Do not select **Any Available**. **Note:** The target domain must contain the same type of storage to import the image. ■ At least one operation must be a **Replication** operation to a **Media Server Deduplication Pool** in another NetBackup domain. You can configure multiple **Replication** operations in an Auto Image Replication SLP. The **Replication** operation settings determine whether the backup is replicated to all replication targets in all primary server domains or only to specific replication targets. ■ The SLP must be of the same data classification as the **Import** SLP in Domain 2.

Table 4-33 SLP requirements for Auto Image Replication *(continued)*

Domain	Storage lifecycle policy requirements
Domain 2 (Target domain)	If replicating to all targets in all domains, in each domain NetBackup automatically creates an **Import** SLP that meets all the necessary criteria. **Note:** If replicating to specific targets, you must create the **Import** SLP before creating the Auto Image Replication SLP in the originating domain. The **Import** SLP must meet the following criteria: ■ The first operation in the SLP must be an **Import** operation. NetBackup must support the **Destination storage** as a target for replication from the source storage. Indicate the exact storage unit from the drop-down list. Do not select **Any Available**. ■ The SLP must contain at least one operation that has the **Target retention** specified. ■ The SLP must be of the same data classification as the SLP in Domain 1. Matching the data classification keeps a consistent meaning to the classification and facilitates global reporting by data classification.

Figure 4-9 shows how the SLP in the target domain is set up to replicate the images from the originating primary server domain.

Figure 4-9 Storage lifecycle policy pair required for Auto Image Replication



Note: Restart `nbstserv` after you make changes to the underlying storage for any operation in an SLP.

Creating a storage lifecycle policy

A storage lifecycle policy (SLP) is a storage plan for a set of backups. The operations in an SLP are the backup instructions for the data. Use the following procedure to create an SLP that contains multiple storage operations.

To add a storage operation to a storage lifecycle policy

1. On the left, select **Storage > Storage lifecycle policies**.
2. Click **Add**.
3. Enter the Storage lifecycle policy name.

   See "Storage lifecycle policy settings" on page 166.
4. Add one or more operations to the SLP. The operations are the instructions for the SLP to follow and apply to the data that is specified in the backup policy.

   If this is the first operation added to the SLP, click **Add**.

   To add a child operation, select an operation and then click **Add child**.
5. Select an **Operation** type. If you're creating a child operation, the SLP displays only those operations that are valid based on the parent operation that you selected.
6. Configure the properties for the operation.
7. The **Window** tab displays for the following operation types: **Backup From Snapshot**, **Duplication**, **Import**, **Index From Snapshot**, and **Replication**. If you'd like to control when the secondary operation runs, create a window for the operation.
8. On the **Properties** tab, go to and expand **Advanced**. Choose if NetBackup should process active images after the window closes.
9. Select **Create** to create the operation.
10. Add additional operations to the SLP as needed. (See step 4.)
11. Change the hierarchy of the operations in the SLP if necessary.
12. Select **Create** to create the SLP. NetBackup validates the SLP when it is first created and whenever it is changed.
13. Configure a backup policy and select a storage lifecycle policy as the **Policy storage**.

Storage lifecycle policy settings

The following table describes the settings for storage lifecycle policies.

Figure 4-10 Storage lifecycle policy tab

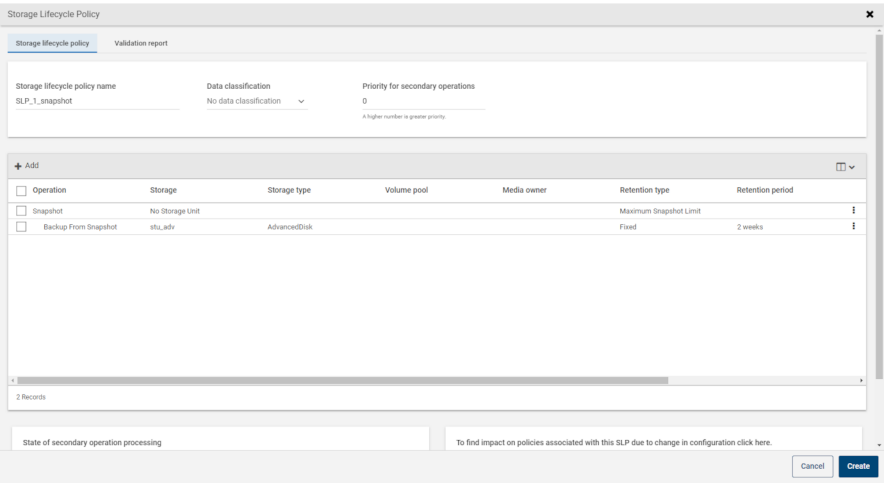


Table 4-34 Storage lifecycle policy tab

Setting	Description
Storage lifecycle policy name	The **Storage lifecycle policy name** describes the SLP. The name cannot be modified after the SLP is created.

Table 4-34 Storage lifecycle policy tab *(continued)*

Setting	Description
Data classification	The **Data classification** defines the level or classification of data that the SLP is allowed to process. The dropdown menu contains all of the defined classifications as well as the **Any** classification, which is unique to SLPs. The **Any** selection indicates to the SLP that it should preserve all images that are submitted, regardless of their data classification. This setting is available for SLP configuration only and is not available to configure a backup policy. In an Auto Image Replication configuration where the master server domains run different versions of NetBackup, see the following topic for special considerations: See "About the storage lifecycle policies required for Auto Image Replication" on page 163. The **Data classification** is an optional setting. One data classification can be assigned to each SLP and applies to all operations in the SLP. If a data classification is selected (other than **Any**), the SLP stores only those images from the policies that are set up for that data classification. If no data classification is indicated, the SLP accepts images of any classification or no classification. The **Data classification** setting allows the NetBackup administrator to classify data based on relative importance. A classification represents a set of backup requirements. When data must meet different backup requirements, consider assigning different classifications. For example, email backup data can be assigned to the silver data classification and financial data backup may be assigned to the platinum classification. A backup policy associates backup data with a data classification. Policy data can be stored only in an SLP with the same data classification. Once data is backed up in an SLP, the data is managed according to the SLP configuration. The SLP defines what happens to the data from the initial backup until the last copy of the image has expired.
Priority for secondary operations	The **Priority for secondary operations** option is the priority that jobs from secondary operations have in relationship to all other jobs. The priority applies to the jobs that result from all operations except for **Backup** and **Snapshot** operations. Range: 0 (default) to 99999 (highest priority). For example, you may want to set the **Priority for secondary operations** for a policy with a gold data classification higher than for a policy with a silver data classification. The priority of the backup job is set in the backup policy on the **Attributes** tab.

Table 4-34 Storage lifecycle policy tab *(continued)*

Setting	Description
Operations	Use the **Add child**, **Edit**, and **Delete** options to create a list of operations in the SLP. An SLP must contain one or more operations. Multiple operations imply that multiple copies are created. The list also contains the columns that display information about each operation. Not all columns are displayed by default.
Move	Use the move option to change the hierarchy of the source for each copy. One copy can be the source for many other copies.
State of Secondary Operation Processing	The **Active** and **Postponed** options refer to the processing of all duplication operations in the SLP. **Note:** These options apply to any duplication operations that create tar-formatted images. For example, those created with `bpduplicate`. These options do not affect the images that are duplicated as a result of OpenStorage optimized duplication or NDMP. Nor do they affect if one or more destination storage units are specified as part of a storage unit group. ■ Enable **Active** to let secondary operations continue as soon as possible. When an operation is changed from **Postponed** to **Active**, NetBackup continues to process the images, picking up where it left off when secondary operations were made inactive. ■ Enable **Postponed** to postpone the secondary operations for the entire SLP. **Postponed** does not postpone the creation of duplication jobs, it postpones the creation of images instead. The duplication jobs continue to be created, but they are not run until secondary operations are active again. All secondary operations in the SLP are inactive indefinitely unless the administrator selects **Active** or until the **Until** option is selected and an activation date is indicated.
Check for conflicts with backup policies option	Select this option to see how changes to this SLP can affect the policies that are associated with this SLP. The option generates a report that displays on the **Validation report** tab. This option performs the same validation as the `-conflict` option performs when used with the `nbstl` command.

About MSDP backup policy configuration

When you configure a backup policy, for the **Policy storage** select a storage unit that uses a deduplication pool.

For a storage lifecycle policy, for the **Storage unit** select a storage unit that uses a deduplication pool.

For VMware backups, select the **Enable file recovery from VM backup** option when you configure a VMware backup policy. The **Enable file recovery from VM backup** option provides the best deduplication rates.

NetBackup deduplicates the client data that it sends to a deduplication storage unit.

See "About storage unit groups for MSDP" on page 54.

See "Use MSDP compression and encryption" on page 53.

Resilient network properties

To access this setting, in the web UI select **Hosts > Host properties**. Select the server or client. If necessary click **Connect**, then click **Edit primary server**, **Edit media server**, or **Edit client**. Click **Resilient network**.

For media servers and clients, the **Resilient network** properties are read only. When a job runs, the primary server updates the media server and the client with the current properties.

The **Resilient network** properties let you configure NetBackup to use resilient network connections for backups and restores. A resilient connection allows backup and restore traffic between a client and a NetBackup media server to function effectively in high-latency, low-bandwidth networks such as WANs. The data travels across a wide area network (WAN) to media servers in a central datacenter.

NetBackup monitors the socket connections between the remote client and the NetBackup media server. If possible, NetBackup re-establishes dropped connections and resynchronizes the data stream. NetBackup also overcomes latency issues to maintain an unbroken data stream. A resilient connection can survive network interruptions of up to 80 seconds. A resilient connection may survive interruptions longer than 80 seconds.

The NetBackup Remote Network Transport Service manages the connection between the computers. The Remote Network Transport Service runs on the primary server, the client, and the media server that processes the backup or restore job. If the connection is interrupted or fails, the services attempt to re-establish a connection and synchronize the data.

NetBackup protects only the network socket connections that the NetBackup Remote Network Transport Service (`nbrntd`) creates. Examples of the connections that are not supported are:

- Clients that back up their own data (deduplication clients and SAN clients)
- Granular Recovery Technology (GRT) for Exchange Server or SharePoint Server
- NetBackup `nbfsd` process.

NetBackup protects connections only after they are established. If NetBackup cannot create a connection because of network problems, there is nothing to protect.

Resilient connections apply between clients and NetBackup media servers, which includes primary servers when they function as media servers. Resilient connections do not apply to primary servers or media servers if they function as clients and back up data to a media server.

Resilient connections can apply to all of the clients or to a subset of clients.

Note: If a client is in a subdomain that is different from the server subdomain, add the fully qualified domain name of the server to the client's hosts file. For example, `india.domain.org` is a different subdomain than `china.domain.org`.

When a backup or restore job for a client starts, NetBackup searches the **Resilient network** list from top to bottom looking for the client. If NetBackup finds the client, NetBackup updates the resilient network setting of the client and the media server that runs the job. NetBackup then uses a resilient connection.

Table 4-35 Resilient network properties

Property	Description
FQDN or IP address	The full qualified domain name or IP address of the host. The address can also be a range of IP addresses so you can configure more than one client at once. You can mix IPv4 addresses and ranges with IPv6 addresses and subnets. If you specify the host by name, it is recommended that you use the fully qualified domain name. Use the arrow buttons on the right side of the pane to move up or move down an item in the list of resilient networks.
Resiliency	**Resiliency** is either **On** or **Off**.

Note: The order is significant for the items in the list of resilient networks. If a client is in the list more than once, the first match determines its resilient connection status. For example, suppose you add a client and specify the client IP address and specify **On** for **Resiliency**. Suppose also that you add a range of IP addresses as **Off**, and the client IP address is within that range. If the client IP address appears before the address range, the client connection is resilient. Conversely, if the IP range appears first, the client connection is not resilient.

Other NetBackup properties control the order in which NetBackup uses network addresses.

The NetBackup resilient connections use the SOCKS protocol version 5.

Resilient connection traffic is not encrypted. It is recommended that you encrypt your backups. For deduplication backups, use the deduplication-based encryption. For other backups, use policy-based encryption.

Resilient connections apply to backup connections. Therefore, no additional network ports or firewall ports must be opened.

Note: If multiple backup streams run concurrently, the Remote Network Transport Service writes a large amount of information to the log files. In such a scenario, it is recommended that you set the logging level for the Remote Network Transport Service to 2 or less. Instructions to configure unified logs are in a different guide.

Resilient connection resource usage

Resilient connections consume more resources than regular connections, as follows:

- More socket connections are required per data stream. Three socket connections are required to accommodate the Remote Network Transport Service that runs on both the media server and the client. Only one socket connection is required for a non-resilient connection.
- More sockets are open on media servers and clients. Three open sockets are required rather than one for a non-resilient connection. The increased number of open sockets may cause issues on busy media servers.
- More processes run on media servers and clients. Usually, only one more process per host runs even if multiple connections exist.
- The processing that is required to maintain a resilient connection may reduce performance slightly.

Specify resilient connections for clients

Use the following procedure to specify resilient connections for NetBackup clients.

See "Resilient network properties" on page 170.

Alternatively, you can use the `resilient_clients` script to specify resilient connections for clients:

- Windows: *`install_path`*`\NetBackup\bin\admincmd\resilient_clients`
- UNIX: `/usr/openv/netbackup/bin/admincmd/resilient_clients`

To specify resilient connections for clients

1. Open the **NetBackup web UI**.
2. On the left, click **Hosts > Host properties**.

3 Select the primary server. If necessary, click **Connect**. Then click **Edit primary server**.

4 Click **Resilient network**.

5 You can perform the following actions:

Add a setting

To add a host or IP address setting

1 Click **Add**.

2 Enter a client host name or an IP address.

If you specify the client host by name, it is recommended that you use the fully qualified domain name.

3 Ensure that the **On** option is selected.

4 Click **Add and add another**.

5 Repeat until you have added each setting.

6 When you finish adding network settings, click **Add**.

Edit a setting

To edit a host or IP address setting

1 Locate the client host name or the IP address.

2 Click **Actions > Edit**.

3 Select the desired **Resiliency** setting.

4 Click **Save**.

Delete a setting

Delete a host or IP address setting

1 Locate the client host name or the IP address.

2 Click **Actions > Delete**.

Up arrow, Down arrow

Change the order of items

1 Select the client host name or the IP address.

2 Click the **Up** or **Down** button.

The order of the items in the list is significant.

See "Resilient network properties" on page 170.

The settings are propagated to the affected hosts through normal NetBackup inter-host communication, which can take up to 15 minutes.

6 If you want to begin a backup immediately, restart the NetBackup services on the primary server.

Adding an MSDP load balancing server

You can add a load balancing server to an existing media server deduplication node.

See "About MSDP storage servers" on page 35.

To add a load balancing server

1. Open the web UI.
2. On the left, click **Storage > Disk storage**.
3. Click the **Storage servers** tab.
4. Click on the deduplication storage server.
5. Under **Media servers**, click **Add**.
6. Select the media servers that you want to use as load balancing servers and click **Add**.

About variable-length deduplication on NetBackup clients

Currently, NetBackup deduplication follows a fixed-length deduplication method where the data streams are chunked into fixed-length segments (128 KB) and then processed for deduplication. Fixed-length deduplication has the advantage of being a swift method and it consumes less computing resources. Fixed-length deduplication handles most kinds of data streams efficiently. However, there can be cases where fixed-length deduplication might result in low deduplication ratios.

If your data was modified in a shifting mode, that is, if some data was inserted in the middle of a file, then variable-length deduplication enables you to get higher deduplication ratios when you back up the data. Variable-length deduplication reduces backup storage, improves the backup performance, and lowers the overall cost that is spent on data protection.

Note: Use variable-length deduplication for data types that do not show a good deduplication ratio with the current MSDP intelligent deduplication algorithm and affiliated streamers. Enabling Variable-length deduplication might improve the deduplication ratio, but consider that the CPU performance might get affected.

In variable-length deduplication, every segment has a variable size with configurable size boundaries. The NetBackup client examines and applies a secure hash algorithm (SHA-2) to the variable-length segments of the data. Each data segment

is assigned a unique ID and NetBackup evaluates if any data segment with the same ID exists in the backup. If the data segment already exists, then the segment data is not stored again.

Warning: If you enable compression for the backup policy, variable-length deduplication does not work even when you configure it.

The following table describes the effect of variable-length deduplication on the data backup:

Table 4-36 Effect of variable-length deduplication

Effect on the deduplication ratio	Variable-length deduplication is beneficial if the data file is modified in a shifting mode, that is when data is inserted, removed, or modified at a binary level. When such modified data is backed up again, variable-length deduplication achieves a higher deduplication ratio. Thus, the second or subsequent backups have higher deduplication ratios.
Effect on the CPU	Variable-length deduplication can be a bit more resource-intensive than fixed-length deduplication to achieve a better deduplication ratio. Variable-length deduplication needs more CPU cycles to compute segment boundaries and the backup time might be more than the fixed-length deduplication method.
Effect on data restore	Variable-length deduplication does not affect the data restore process.

Configure variable-length deduplication

By default, the variable-length deduplication is disabled on a NetBackup client. From NetBackup 10.2 onwards, you can enable variable-length deduplication by using **cacontrol** command-line utility. In the previous versions of NetBackup, you can enable it by adding parameters in the `pd.conf` file. To enable the same settings for all NetBackup clients or policies, you must specify all the clients or policies in the `pd.conf` file.

From NetBackup 10.2 onwards, the default version of the variable-length deduplication is VLD v2. If you have enabled variable-length deduplication in `pd.conf` file and image backups do not exists in the storage, VLD v2 is used by default. If image backups already exist in the storage, NetBackup continues to use VLD v1.

In a deduplication load balancing scenario, you must upgrade the media servers to NetBackup 8.1.1 or later and modify the `pd.conf` file on all the media servers. If a backup job selects an older media server (earlier than NetBackup 8.1.1) for the load balancing pool, fixed-length deduplication is used instead of variable-length deduplication. Avoid configuring media servers with different NetBackup versions in a load balancing scenario. The data segments generated from variable-length deduplication are different from the data segments generated from fixed-length deduplication. Therefore, load balancing media servers with different NetBackup versions results in a low deduplication ratio.

See "Managing the variable-length deduplication using the cacontrol command-line utility" on page 176.

See "About the MSDP pd.conf configuration file" on page 178.

See "Editing the MSDP pd.conf file" on page 179.

See "MSDP pd.conf file parameters" on page 179.

Managing the variable-length deduplication using the cacontrol command-line utility

You can use **cacontrol** command-line utility to set the variable-length deduplication. Use `--vld` flag to set the variable-length deduplication by creating the marker entry for a client or policy in a configuration file.

The **cacontrol** command-line utility is located at the following locations:

- Windows: `install_path\Veritas\pdde\cacontrol`
- UNIX: `/usr/openv/pdde/pdcr/bin/cacontrol`

From NetBackup 10.2 and later, you can configure variable-length deduplication using the following options.

Table 4-37 Command options for VLD in **cacontrol**

Option	Description
`client`	Enable Variable Length Deduplication for a specific client.
`policy`	Enable Variable Length Deduplication for specific policy.

Table 4-37 Command options for VLD in **cacontrol** *(continued)*

Option	Description
`vldtype`	■ `VLD` The version 1 of the variable-length deduplication algorithm. ■ `VLD_2` The version 2 of the variable-length deduplication algorithm. It is recommended to use this version as default. ■ `VLD_3` Another version of the variable-length deduplication algorithm.
`minsegsize`	The minimum segment size (KB) of the segmentation range for Variable Length Deduplication segmentation (KB). Suggested values are 16, 32, and 64. Sizes must be a multiple of 4 and fall in the range 4-16384.
`maxsegSize`	The maximum segment size (KB) of the segmentation range for Variable Length Deduplication segmentation (KB), this value must be greater than `sw_min`. Suggested values are 32, 64, and 128. The maximum segment size must be greater than the minimum.

Warning: A change in a variable-length deduplication version decreases the deduplication ratio of image backups until new images are backed up once or twice. Therefore, choose new variable-length deduplication cautiously.

To manage variable-length deduplication for MSDP by using cacontrol command-line utility

1 Query the active settings for the client and policy.

```
cacontrol --vld queryactive <CLIENT> <POLICY>
```

2 Set the configuration for client and policy.

```
cacontrol --vld update <CLIENT> <POLICY> <VLDTYPE>
<MINSEGMENTSIZE> <MAXSEGMENTSIZE>
```

3 Delete the settings for client and policy.

```
cacontrol --vld delete <CLIENT> <POLICY>
```

4 Query the settings for client and policy.

```
cacontrol --vld get <CLIENT> <POLICY>
```

5 Set the configuration for the policy.

```
cacontrol –-vld updatebypolicy <POLICY> <VLDTYPE> <MINSEGMENTSIZE>
<MAXSEGMENTSIZE>
```

6 Delete the settings for client.

```
cacontrol --vld deletebypolicy <POLICY>
```

7 Query the settings for the policy.

```
cacontrol --vld getbypolicy <POLICY>
```

8 Set the configuration for the client.

```
cacontrol –-vld updatebyclient <CLIENT> <VLDTYPE> <MINSEGMENTSIZE>
<MAXSEGMENTSIZE>
```

9 Delete the settings for client.

```
cacontrol --vld deletebyclient <CLIENT>
```

10 Query the settings for the client.

```
cacontrol --vld getbyclient <CLIENT>
```

When you set parameters for client and policy, you can use asterisk (*) to indicate all clients or policies.

For example,

```
cacontrol --vld updatebypolicy "*" VLD_V2 32 64
```

About the MSDP pd.conf configuration file

On each NetBackup host that deduplicates data, a `pd.conf` file contains the various configuration settings that control the operation of deduplication for the host. By default, the `pd.conf` file settings on the deduplication storage server apply to all load balancing servers and all clients that deduplicate their own data.

You can edit the file to configure advanced settings for that host. If a configuration setting does not exist in a `pd.conf` file, you can add it. If you change the `pd.conf` file on a host, it changes the settings for that host only. If you want the same settings for all of the hosts that deduplicate data, you must change the `pd.conf` file on all of the hosts.

The `pd.conf` file settings may change between releases. During upgrades, NetBackup adds only the required settings to existing `pd.conf` files.

The `pd.conf` file resides in the following directories:

- (UNIX) `/usr/openv/lib/ost-plugins/`

- (Windows) *install_path*\Veritas\NetBackup\bin\ost-plugins

See "MSDP pd.conf file parameters" on page 179.

See "Editing the MSDP pd.conf file" on page 179.

Editing the MSDP pd.conf file

If you change the pd.conf file on a host, it changes the settings for that host only. If you want the same settings for all of the hosts that deduplicate data, you must change the pd.conf file on all of the hosts.

Note: Cohesity recommends that you make a backup copy of the file before you edit it.

See "About the MSDP pd.conf configuration file" on page 178.

See "MSDP pd.conf file parameters" on page 179.

To edit the pd.conf file

1. Use a text editor to open the pd.conf file.

   The pd.conf file resides in the following directories:

   - (UNIX) /usr/openv/lib/ost-plugins/
   - (Windows) *install_path*\Veritas\NetBackup\bin\ost-plugins

2. To activate a setting, remove the pound character (#) in column 1 from each line that you want to edit.
3. To change a setting, specify a new value.

   Note: The spaces to the left and right of the equal sign (=) in the file are significant. Ensure that the space characters appear in the file after you edit the file.

4. Save and close the file.
5. Restart the NetBackup Remote Manager and Monitor Service (nbrmms) on the host.

MSDP pd.conf file parameters

Table 4-38 describes the deduplication parameters that you can configure for a NetBackup **Media Server Deduplication Pool** environment.

The parameters in this table are in alphabetical order; the parameters in a `pd.conf` file may not be in alphabetical order.

The parameters in the file in your release may differ from those that are described in this topic.

You can edit the file to configure advanced settings for a host. If a parameter does not exist in a `pd.conf` file, you can add it. During upgrades, NetBackup adds only required parameters to existing `pd.conf` files.

The `pd.conf` file resides in the following directories:

- (Windows) *`install_path`*`\Veritas\NetBackup\bin\ost-plugins`
- (UNIX) `/usr/openv/lib/ost-plugins/`

Table 4-38 pd.conf file parameters

Parameter	Description
`BACKUPRESTORERANGE`	On a client, specifies the IP address or range of addresses that the local network interface card (NIC) should use for backups and restores. Specify the value in one of two ways, as follows: ■ Classless Inter-Domain Routing (CIDR) format. For example, the following notation specifies 192.168.10.0 and 192.168.10.1 for traffic: `BACKUPRESTORERANGE = 192.168.10.1/31` ■ Comma-separated list of IP addresses. For example, the following notation specifies 192.168.10.1 and 192.168.10.2 for traffic: `BACKUPRESTORERANGE = 192.168.10.1, 192.168.10.2` Default value: `BACKUPRESTORERANGE=` (no default value) Possible values: Classless Inter-Domain Routing format notation or comma-separated list of IP addresses
`BANDWIDTH_LIMIT`	Determines the maximum bandwidth that is allowed when backing up or restoring data between the deduplication host and the deduplication pool. The value is specified in KBytes/second. The default is no limit. Default value: `BANDWIDTH_LIMIT = 0` Possible values: `0` (no limit) to the practical system limit, in KBs/sec
`COMPRESSION`	Specifies whether to compress the data during backups. By default, the data is compressed. Default value: `COMPRESSION = 1` Possible values: `0` (off) or `1` (on) See "About MSDP compression" on page 108.

Table 4-38 pd.conf file parameters *(continued)*

Parameter	Description
CR_STATS_TIMER	Specifies a time interval in seconds for retrieving statistics from the storage server host. The default value of 0 disables caching and retrieves statistics on demand. Consider the following information before you change this setting: ■ If disabled (set to 0), a request for the latest storage capacity information occurs whenever NetBackup requests it. ■ If you specify a value, a request occurs only after the specified number of seconds since the last request. Otherwise, a cached value from the previous request is used. ■ Enabling this setting may reduce the queries to the storage server. The drawback is the capacity information reported by NetBackup becomes stale. Therefore, if storage capacity is close to full, Cohesity recommends that you do not enable this option. ■ On high load systems, the load may delay the capacity information reporting. If so, NetBackup may mark the storage unit as down. Default value: CR_STATS_TIMER = 0 Possible values: 0 or greater, in seconds **Note:** Do not configure the CR_STATS_TIMER parameter in pd.conf file if msdpcloud is configured in the environment.
DEBUGLOG	Specifies the file to which NetBackup writes the deduplication plug-in log information. NetBackup prepends a date stamp to each day's log file. On Windows, a partition identifier and slash must precede the file name. On UNIX, a slash must precede the file name. **Note:** This parameter does not apply for NDMP backups from a NetApp appliance. Default value: ■ UNIX: DEBUGLOG = /var/log/puredisk/pdplugin.log ■ Windows: DEBUGLOG = C:\pdplugin.log Possible values: Any path

Table 4-38 pd.conf file parameters *(continued)*

Parameter	Description
DISABLE_BACKLEVEL_TLS	When secure communication is established between the client and the server, this parameter specifies whether or not to disable older TLS versions. NetBackup version 8.0 and earlier use older TLS versions such as SSLV2, SSLV3, TLS 1.0, and TLS 1.1. To enable TLS 1.2, change the value of the DISABLE_BACKLEVEL_TLS parameter to 1 and restart the NetBackup Deduplication Engine (spoold) and the NetBackup Deduplication Manager (spad). Default value: `DISABLE_BACKLEVEL_TLS = 0` Possible values: 0 (off) or 1 (on) **Note:** To enable TLS 1.2, NetBackup version must be 8.1 and later. When TLS 1.2 is enabled (`DISABLE_BACKLEVEL_TLS = 1`) on a machine (which can be a client or a media server or a load balance server), to establish communication, all machines connected to it must also enable TLS 1.2. For a standard backup, NetBackup client version 8.0 and earlier can communicate with NetBackup server (media server or load balance server) version 8.1 that has TLS 1.2 enabled. However, in case of optimized duplication and replication, load balance, and client direct duplication, NetBackup client versions 8.0 and earlier cannot communicate with NetBackup server (media server or load balance server) version 8.1, which has TLS 1.2 enabled.
DONT_SEGMENT_TYPES	A comma-separated list of file name extensions of files not to be deduplicated. Files in the backup stream that have the specified extensions are given a single segment if smaller than 16 MB. Larger files are deduplicated using the maximum 16-MB segment size. Example: `DONT_SEGMENT_TYPES = mp3,avi` This setting prevents NetBackup from analyzing and managing segments within the file types that do not deduplicate globally. Note: this parameter does not apply to the NDMP backups that use the NetApp stream handler. Default value: `DONT_SEGMENT_TYPES =` (no default value) Possible values: comma-separated file extensions

Table 4-38 pd.conf file parameters *(continued)*

Parameter	Description
ENCRYPTION	Specifies whether to encrypt the data during backups. By default, files are not encrypted. If you set this parameter to 1 on all hosts, the data is encrypted during transfer and on the storage. Default value: ENCRYPTION = 0 Possible values: 0 (no encryption) or 1 (encryption) See "About MSDP encryption" on page 109. To encrypt all data in the MSDP server, it is recommended that you use the server option. ENCRYPTION parameter is useful only for the backups or replication using the hosts where the pd.conf file exists.
FIBRECHANNEL	Enables the Fibre Channel for backup, and restores the traffic to and from a NetBackup series appliance. Default value: FIBRECHANNEL = 0 Possible values: 0 (off) or 1 (on)

Table 4-38 pd.conf file parameters *(continued)*

Parameter	Description
FILE_KEEP_ALIVE_INTERVAL	The interval in seconds at which to perform keepalives on idle sockets. The following items describe the behavior based on how you configure this parameter: ■ Commented out (default) and Resilient Network connections are enabled: If the value is less than 75 seconds, the keep alive interval is 60 seconds. If the value is greater than 1800 seconds (30 minutes), the keep alive interval is 1440 seconds (80% of 30 minutes). If the value is between 75 and 1800 sections, the keep-alive interval is 80% of the parameter value. See "Resilient network properties" on page 170. ■ Commented out (the default) and Resilient Network connections are *not* enabled: The keep-alive interval is 1440 seconds (80% of 30 minutes). ■ 0 or less: Disabled; no keepalives are sent. ■ Greater than 0: The keep-alive interval is the specified value in seconds except as follows: If less than 60 seconds or greater than 7200 seconds (two hours), the keep-alive interval is 1440 seconds (80% of 30 minutes). Default value : `FILE_KEEP_ALIVE_INTERVAL = 1440` Possible values: `0` (disabled) or `60` to `7200` seconds To determine the keep alive interval that NetBackup uses, examine the deduplication plug-in log file for a message similar to the following: `Using keepalive interval of xxxx seconds` For more information about the deduplication plug-in log file, see `DEBUGLOG` and `LOGLEVEL` in this table.

Table 4-38 pd.conf file parameters *(continued)*

Parameter	Description
FP_CACHE_CLIENT_POLICY	**Note:** Cohesity recommends that you use this setting on the individual clients that back up their own data (client-side deduplication). If you use it on a storage server or load balancing server, it affects all backup jobs. Specifies the client, backup policy, and date from which to obtain the fingerprint cache for the first backup of a client. By default, the fingerprints from the previous backup are loaded. This parameter lets you load the fingerprint cache from another, similar backup. It can reduce the amount of time that is required for the first backup of a client. This parameter especially useful for remote office backups to a central datacenter in which data travels long distances over a WAN. Specify the setting in the following format: ***clienthostmachine,backuppolicy,date*** The ***date*** is the last date in mm/dd/yyyy format to use the fingerprint cache from the client you specify. Default value: `FP_CACHE_CLIENT_POLICY =` (no default value) See "Configuring MSDP fingerprint cache seeding on the client" on page 83.
FP_CACHE_INCREMENTAL	Specifies whether to use fingerprint caching for incremental backups. Because incremental backups only back up what has changed since the last backup, cache loading has little affect on backup performance for incremental backups. Default value: `FP_CACHE_INCREMENTAL = 0` Possible values: `0` (off) or `1` (on) **Note:** Change this value only when directed to do so by a Cohesity representative.
FP_CACHE_LOCAL	Specifies whether or not to use the fingerprint cache for the backup jobs that are deduplicated on the storage server. This parameter does not apply to load balancing servers or to clients that deduplicate their own data. When the deduplication job is on the same host as the NetBackup Deduplication Engine, disabling the fingerprint cache improves performance. Default value: `FP_CACHE_LOCAL = 1` Possible values: `0` (off) or `1` (on)

Table 4-38 pd.conf file parameters *(continued)*

Parameter	Description
FP_CACHE_MAX_COUNT	Specifies the maximum number of images to load in the fingerprint cache. Default value: FP_CACHE_MAX_COUNT = 1024 Possible values: 0 to 4096 **Note:** Change this value only when directed to do so by a Cohesity representative.
FP_CACHE_MAX_MBSIZE	Specifies the amount of memory in MBs to use for the fingerprint cache. Default value: FP_CACHE_MAX_MBSIZE = 20 Possible values: 0 to the computer limit **Note:** Change this value only when directed to do so by a Cohesity representative.
FP_CACHE_PERIOD_REBASING_THRESHOLD	Specifies the threshold (MB) for periodic rebasing during backups. A container is considered for rebasing if both of the following are true: ■ The container has not been rebased within the last three months. ■ For that backup, the data segments in the container consume less space than the FP_CACHE_PERIOD_REBASING_THRESHOLD value. Default value: FP_CACHE_PERIOD_REBASING_THRESHOLD = 16 Possible values: 0 (disabled) to 256 See "About MSDP storage rebasing" on page 561.
FP_CACHE_REBASING_THRESHOLD	Specifies the threshold (MB) for normal rebasing during backups. A container is considered for rebasing if both of the following are true: ■ The container has been rebased within the last three months. ■ For that backup, the data segments in the container consume less space than the FP_CACHE_REBASING_THRESHOLD value. Default value:FP_CACHE_REBASING_THRESHOLD = 4 Possible values: 0 (disabled) to 200 If you change this value, consider the new value carefully. If you set it too large, all containers become eligible for rebasing. Deduplication rates are lower for the backup jobs that perform rebasing. See "About MSDP storage rebasing" on page 561.

Table 4-38 pd.conf file parameters *(continued)*

Parameter	Description
LOCAL_SETTINGS	Specifies whether to use the `pd.conf` settings of the local host or to allow the server to override the local settings. The following is the order of precedence for local settings: ■ Local host ■ Load balancing server ■ Storage server To use the local settings, set this value to `1`. Default value: `LOCAL_SETTINGS = 0` Possible values: `0` (allow override) or `1` (always use local settings)
LOGLEVEL	Specifies the amount of information that is written to the log file. The range is from 0 to 10, with 10 being the most logging. Default value: `LOGLEVEL = 0` Possible values: An integer, `0` to `10` inclusive **Note:** Change this value only when directed to do so by a Cohesity representative.
MAX_IMG_MBSIZE	The maximum backup image fragment size in megabytes. Default value: `MAX_IMG_MBSIZE = 51200` Possible values: `0` to `51,200`, in MBs **Note:** Change this value only when directed to do so by a Cohesity representative.
MAX_LOG_MBSIZE	The maximum size of the log file in megabytes. NetBackup creates a new log file when the log file reaches this limit. NetBackup prepends the date and the ordinal number beginning with 0 to each log file, such as `120131_0_pdplugin.log`, `120131_1_pdplugin.log`, and so on. Default value: `MAX_LOG_MBSIZE = 100` Possible values: `0` to `50,000`, in MBs
META_SEGKSIZE	The segment size for metadata streams Default value: `META_SEGKSIZE = 16384` Possible values: 32-16384, multiples of 32 **Note:** Change this value only when directed to do so by a Cohesity representative.

Table 4-38 pd.conf file parameters *(continued)*

Parameter	Description
MTSTRM_BACKUP_CLIENTS	If set, limits the use of the Multi-Threaded Agent to the backups of the specified clients. The clients that are not specified use single-threading. This setting does not guarantee that the specified clients use the Multi-Threaded Agent. The MaxConcurrentSessions parameter in the mtstrm.conf file controls the number of backups the Multi-Threaded Agent processes concurrently. If you specify more clients than the MaxConcurrentSessions value, some of the clients may use single-threaded processing. See "MSDP mtstrm.conf file parameters" on page 68. The format is a comma-separated list of the clients, case insensitive (for example, MTSTRM_BACKUP_CLIENTS = client1,client2,client3). Default value: MTSTRM_BACKUP_CLIENTS = (no default value) Possible values: comma separated client names See "About the MSDP Deduplication Multi-Threaded Agent" on page 66.

Table 4-38 pd.conf file parameters *(continued)*

Parameter	Description
MTSTRM_BACKUP_ENABLED	Use the Multi-Threaded Agent in the backup stream between the deduplication plug-in and the NetBackup Deduplication Engine. Default value: MTSTRM_BACKUP_ENABLED = (no default value) Possible values: 1 (On) or 0 (Off) The value for this parameter is configured during installation or upgrade. If the hardware concurrency value of the host is greater than a hardware concurrency threshold value, NetBackup sets MTSTRM_BACKUP_ENABLED to 1. (For the purposes of this parameter, the *hardware concurrency* is the number of CPUs or cores or hyperthreading units.) The following items describe the values that are used for the determination algorithm: ■ The hardware concurrency value is one of the following: ■ For media servers, half of the host's hardware concurrency is used for the hardware concurrency value in the algorithm. ■ For clients, all of the host's hardware concurrency is used for the hardware concurrency value in the algorithm. ■ The hardware concurrency threshold value to enable multithreading is one of the following: ■ Windows and Linux: The threshold value is 2. ■ Solaris: The threshold value is 4. The following examples may be helpful: ■ A Linux media server that has 8 CPU cores with two hyperthreading units per core has a hardware concurrency of 16. Therefore, the hardware concurrency value for the algorithm is 8 (for media servers, half of the system's hardware concurrency). Eight is greater than two (the threshold value of Windows and Linux), so multithreading is enabled (MTSTRM_BACKUP_ENABLED = 1). ■ A Solaris client that has 2 CPU cores without hyperthreading has a hardware concurrency of 2. The hardware concurrency value for the algorithm is 2 (for clients, all of the system's hardware concurrency). Two is not greater than four (the threshold value of Solaris), so multithreading is not enabled (MTSTRM_BACKUP_ENABLED = 0). See "About the MSDP Deduplication Multi-Threaded Agent" on page 66.

Table 4-38 pd.conf file parameters *(continued)*

Parameter	Description
MTSTRM_BACKUP_POLICIES	If set, limits the use of the Multi-Threaded Agent to the backups of the specified policies. The clients in the policies that are not specified use single-threading, unless the client is specified in the MTSTRM_BACKUP_CLIENTS parameter. This setting does not guarantee that all of the clients in the specified policies use the Multi-Threaded Agent. The MaxConcurrentSessions parameter in the mtstrm.conf file controls the number of backups the Multi-Threaded Agent processes concurrently. If the policies include more clients than the MaxConcurrentSessions value, some of the clients may use single-threaded processing. See "MSDP mtstrm.conf file parameters" on page 68. The format is a comma-separated list of the policies, case sensitive (for example, MTSTRM_BACKUP_POLICIES = policy1,policy2,policy3). Default value: MTSTRM_BACKUP_POLICIES = (no default value) Possible values: comma separated backup policy names See "About the MSDP Deduplication Multi-Threaded Agent" on page 66.
MTSTRM_IPC_TIMEOUT	The number of seconds to wait for responses from the Multi-Threaded Agent before the deduplication plug-in times out with an error. Default value: MTSTRM_IPC_TIMEOUT = 1200 Possible values: 1-86400, inclusive See "About the MSDP Deduplication Multi-Threaded Agent" on page 66.
OPTDUP_BANDWIDTH	Determines the bandwidth that is allowed for each optimized duplication and Auto Image Replication stream on a deduplication server. OPTDUP_BANDWIDTH does not apply to clients. The value is specified in KBytes/second. Default value: OPTDUP_BANDWIDTH= 0 Possible values: 0 (no limit) to the practical system limit, in KBs/sec A global bandwidth parameter effects whether or not OPTDUP_BANDWIDTH applies. See "About configuring MSDP optimized duplication and replication bandwidth" on page 160.

Table 4-38 pd.conf file parameters *(continued)*

Parameter	Description
OPTDUP_COMPRESSION	Specifies whether to compress the data during optimized duplication and Auto Image Replication. By default, files are compressed. To disable compression, change the value to 0. This parameter does not apply to clients. Default value: OPTDUP_COMPRESSION = 1 Possible values: 0 (off) or 1 (on) See "About MSDP compression" on page 108.
OPTDUP_ENCRYPTION	Specifies whether to encrypt the data during optimized duplication and replication. By default, files are not encrypted. If you want encryption, change the value to 1 on the MSDP storage server and on the MSDP load balancing servers. This parameter does not apply to clients. If you set this parameter to 1 on all hosts, the data is encrypted during transfer. Default value: OPTDUP_ENCRYPTION = 0 Possible values: 0 (off) or 1 (on) See "About MSDP encryption" on page 109.
OPTDUP_TIMEOUT	Specifies the number of minutes before the optimized duplication times out. Default value: OPTDUP_TIMEOUT = 720 Possible values: The value, expressed in minutes
PREFERRED_EXT_SEGKSIZE	Specifies the file extensions and the preferred segment sizes in KB for specific file types. File extensions are case sensitive. The following describe the default values: edb are Exchange Server files; mdfare SQL Server master database files, ndf are SQL Server secondary data files, and segsize64k are Microsoft SQL streams. Default value: PREFERRED_EXT_SEGKSIZE = edb:32,mdf:64,ndf:64,segsize64k:64 Possible values: *file_extension*:*segment_size_in_KBs* pairs, separated by commas. See also SEGKSIZE.

Table 4-38 pd.conf file parameters *(continued)*

Parameter	Description
PREFETCH_SIZE	The size in bytes to use for the data buffer for restore operations. Default value: PREFETCH_SIZE = 33554432 Possible values: 0 to the computer's memory limit **Note:** Change this value only when directed to do so by a Cohesity representative.
PREDOWNLOAD_FACTOR	Specifies the predownload factor to use when we restore the data from cloud LSU. Default value: PREDOWNLOAD_FACTOR=40 Possible values: 0 to 100 **Note:** Predownload batch size is PREDOWNLOAD_FACTOR * PREFETCH_SIZE
RESTORE_DECRYPT_LOCAL	Specifies on which host to decrypt and decompress the data during restore operations. Depending on your environment, decryption and decompression on the client may provide better performance. Default value: RESTORE_DECRYPT_LOCAL = 1 Possible values: 0 enables decryption and decompression on the media server; 1 enables decryption and decompression on the client.
SEGKSIZE	The default file segment size in kilobytes. Default value: SEGKSIZE = 128 Possible values: 32 to 16384 KBs, increments of 32 only **Warning:** Changing this value may reduce capacity and decrease performance. Change this value only when directed to do so by a Cohesity representative. You can also specify the segment size for specific file types. See PREFERRED_EXT_SEGKSIZE.

Table 4-38 pd.conf file parameters *(continued)*

Parameter	Description
VLD_CLIENT_NAME	Specifies the name of the NetBackup client to enable variable-length deduplication. By default, the VLD_CLIENT_NAME parameter is not present in the pd.conf configuration file. You can also specify different maximum and minimum segment sizes with this parameter for different NetBackup clients. If you do not specify the segment sizes, then the default values are considered. The values are case-sensitive. Use in any of the following formats: ■ VLD_CLIENT_NAME = * Enables variable-length deduplication for all NetBackup clients and uses the default VLD_MIN_SEGKSIZE and VLD_MAX_SEGKSIZE values. ■ VLD_CLIENT_NAME = *clientname* Enables variable-length deduplication for NetBackup client *clientname* and uses the default VLD_MIN_SEGKSIZE and VLD_MAX_SEGKSIZE values. ■ VLD_CLIENT_NAME = *clientname (64, 256)* Enables variable-length deduplication for NetBackup client *clientname* and uses 64 KB as the VLD_MIN_SEGKSIZE and 256 KB as the VLD_MAX_SEGKSIZE value. **Note:** You can add a maximum of 50 clients in the pd.conf file.
VLD_MIN_SEGKSIZE	The minimum size of the data segment for variable-length deduplication in KB. The segment size must be in multiples of 4 and fall in between 4 KB to 16384 KB. The default value is 64 KB. The value must be smaller than VLD_MAX_SEGKSIZE. Different NetBackup clients can have different segment sizes. A larger value reduces the CPU consumption, but decreases the deduplication ratio. A smaller value increases the CPU consumption, but increases the deduplication ratio **Note:** Keeping similar or close values for VLD_MIN_SEGKSIZE and VLD_MAX_SEGKSIZE results in a performance that is similar to fixed-length deduplication.

Table 4-38 pd.conf file parameters *(continued)*

Parameter	Description
VLD_MAX_SEGKSIZE	The maximum size of the data segment for variable-length deduplication in KB. VLD_MAX_SEGKSIZE is used to set a boundary for the data segments. The segment size must be in multiples of 4 and fall in between 4 KB to 16384 KB. The default value is 128 KB. The value must be greater than VLD_MIN_SEGKSIZE. Different NetBackup clients can have different segment sizes. **Note:** Keeping similar or close values for VLD_MIN_SEGKSIZE and VLD_MAX_SEGKSIZE results in a performance that is similar to fixed-length deduplication.
VLD_POLICY_NAME	Specifies the name of the backup policy to enable variable-length deduplication. By default, the VLD_POLICY_NAME parameter is not present in the pd.conf configuration file. You can also specify different maximum and minimum segment sizes with this parameter for different NetBackup policies. If you do not specify the segment sizes, then the default values are considered. The values are case-sensitive. Use in any of the following formats: ■ VLD_POLICY_NAME = * Enables variable-length deduplication for all NetBackup policies and uses the default VLD_MIN_SEGKSIZE and VLD_MAX_SEGKSIZE values. ■ VLD_POLICY_NAME = *policyname* Enables variable-length deduplication for NetBackup policy *policyname* and uses the default VLD_MIN_SEGKSIZE and VLD_MAX_SEGKSIZE values. ■ VLD_POLICY_NAME = *policyname (64, 256)* Enables variable-length deduplication for NetBackup policy *policyname* and uses 64 KB as the VLD_MIN_SEGKSIZEand 256 KB as the VLD_MAX_SEGKSIZE value.

See "About the MSDP pd.conf configuration file" on page 178.

See "Editing the MSDP pd.conf file" on page 179.

About the MSDP contentrouter.cfg file

The contentrouter.cfg file contains various configuration settings that control some of the operations of your deduplication environment.

Usually, you do not need to change settings in the file. However, in some cases, you may be directed to change settings by a Cohesity support representative.

The NetBackup documentation exposes only some of the `contentrouter.cfg` file parameters. Those parameters appear in topics that describe a task or process to change configuration settings.

Note: Change values in the `contentrouter.cfg` only when directed to do so by the NetBackup documentation or by a Cohesity representative.

The `contentrouter.cfg` file resides in the following directories:

- (UNIX) *storage_path*`/etc/puredisk`
- (Windows) *storage_path*`\etc\puredisk`

See "MSDP server-side rebasing parameters" on page 562.

See "Editing the MSDP pd.conf file" on page 179.

About saving the MSDP storage server configuration

You can save your storage server settings in a text file. A saved storage server configuration file contains the configuration settings for your storage server. It also contains status information about the storage. A saved configuration file may help you with recovery of your storage server. Therefore, Cohesity recommends that you get the storage server configuration and save it in a file. The file does not exist unless you create it.

The following is an example of a populated configuration file:

```
V7.0 "storagepath" "D:\DedupeStorage" string
V7.0 "spalogpath" "D:\DedupeStorage\log" string
V7.0 "dbpath" "D:\DedupeStorage" string
V7.0 "required_interface" "HOSTNAME" string
V7.0 "spalogretention" "7" int
V7.0 "verboselevel" "3" int
V7.0 "replication_target(s)" "none" string
V7.0 "Storage Pool Size" "698.4GB" string
V7.0 "Storage Pool Used Space" "132.4GB" string
V7.0 "Storage Pool Available Space" "566.0GB" string
V7.0 "Catalog Logical Size" "287.3GB" string
V7.0 "Catalog files Count" "1288" string
V7.0 "Space Used Within Containers" "142.3GB" string
```

V7.0 represents the version of the I/O format not the NetBackup release level. The version may differ on your system.

If you get the storage server configuration when the server is not configured or is down and unavailable, NetBackup creates a template file. The following is an example of a template configuration file:

```
V7.0 "storagepath" " " string
V7.0 "spalogin" " " string
V7.0 "spapasswd" " " string
V7.0 "spalogretention" "7" int
V7.0 "verboselevel" "3" int
V7.0 "dbpath" " " string
V7.0 "required_interface" " " string
```

To use a storage server configuration file for recovery, you must edit the file so that it includes only the information that is required for recovery.

See "Saving the MSDP storage server configuration" on page 196.

See "Editing an MSDP storage server configuration file" on page 197.

See "Setting the MSDP storage server configuration" on page 198.

Saving the MSDP storage server configuration

Cohesity recommends that you save the storage server configuration in a file. A storage server configuration file can help with recovery.

See "About saving the MSDP storage server configuration" on page 195.

See "Save the MSDP storage server configuration" on page 55.

See "Recovering from an MSDP storage server disk failure" on page 583.

See "Recovering from an MSDP storage server failure" on page 584.

To save the storage server configuration

- On the primary server, enter the following command:

UNIX: `/usr/openv/netbackup/bin/admincmd/nbdevconfig -getconfig -storage_server sshostname -stype PureDisk -configlist file.txt`

Windows: `install_path\NetBackup\bin\admincmd\nbdevconfig -getconfig -storage_server sshostname -stype PureDisk -configlist file.txt`

For *sshostname*, use the name of the storage server. For *file.txt*, use a file name that indicates its purpose.

If you get the file when a storage server is not configured or is down and unavailable, NetBackup creates a template file.

Editing an MSDP storage server configuration file

To use a storage server configuration file for recovery, it must contain only the required information. You must remove any point–in–time status information. (Status information is only in a configuration file that was saved on an active storage server.) You also must add several configuration settings that are not included in a saved configuration file or a template configuration file.

Table 4-39 shows the configuration lines that are required.

Table 4-39 Required lines for a recovery file

Configuration setting	Description
`V7.0 "storagepath" " " string`	The value should be the same as the value that was used when you configured the storage server.
`V7.0 "spalogpath" " " string`	For the `spalogpath`, use the `storagepath` value and append `log` to the path. For example, if the `storagepath` is `D:\DedupeStorage`, enter `D:\DedupeStorage\log`.
`V7.0 "dbpath" " " string`	If the database path is the same as the `storagepath` value, enter the same value for `dbpath`. Otherwise, enter the path to the database.
`V7.0 "required_interface" " " string`	A value for `required_interface` is required only if you configured one initially; if a specific interface is not required, leave it blank. In a saved configuration file, the required interface defaults to the computer's hostname.
`V7.0 "spalogretention" "7" int`	Do not change this value.
`V7.0 "verboselevel" "3" int`	Do not change this value.
`V7.0 "replication_target(s)" "none" string`	A value for `replication_target(s)` is required only if you configured optimized duplication. Otherwise, do not edit this line.
`V7.0 "spalogin" "`*`username`*`" string`	Replace *username* with the NetBackup Deduplication Engine user ID.
`V7.0 "spapasswd" "`*`password`*`" string`	Replace *password* with the password for the NetBackup Deduplication Engine user ID.
`V7.0 "encryption" " " int`	The value should be the same as the value that was used when you configured the storage server.
`V7.0 "kmsenabled" " " int`	The value is used to enable or disable MSDP KMS configuration. The value should be the same as the value that was used when you configured the storage server.

Table 4-39 Required lines for a recovery file *(continued)*

Configuration setting	Description
`V7.0 "kmsservertype" " " int`	The value is KMS server type. This value should be 0.
`V7.0 "kmsservername" " " string`	The value is NBU Key Management Server. The value should be the same as the value that was used when you configured the storage server. If you use an external KMS as a KMS server, the value must be the NetBackup primary server name. See *External KMS support in NetBackup* in the *NetBackup Security and Encryption Guide*.
`V7.0 "keygroupname" " " string`	The value should be the same as the value that was used when you configured the storage server.

See "About saving the MSDP storage server configuration" on page 195.

See "Recovering from an MSDP storage server disk failure" on page 583.

See "Recovering from an MSDP storage server failure" on page 584.

To edit the storage server configuration

1. If you did not save a storage server configuration file, get a storage server configuration file.

   See "Saving the MSDP storage server configuration" on page 196.

2. Use a text editor to enter, change, or remove values.

   Remove lines from and add lines to your file until only the required lines (see Table 4-39) are in the configuration file. Enter or change the values between the second set of quotation marks in each line. A template configuration file has a space character (`" "`) between the second set of quotation marks.

Setting the MSDP storage server configuration

You can set the storage server configuration (that is, configure the storage server) by importing the configuration from a file. Setting the configuration can help you with recovery of your environment.

See "Recovering from an MSDP storage server disk failure" on page 583.

See "Recovering from an MSDP storage server failure" on page 584.

To set the configuration, you must have an edited storage server configuration file.

See "About saving the MSDP storage server configuration" on page 195.

See "Saving the MSDP storage server configuration" on page 196.

See "Editing an MSDP storage server configuration file" on page 197.

Note: The only time you should use the `nbdevconfig` command with the `-setconfig` option is for recovery of the host or the host disk.

To set the storage server configuration

- On the primary server, run the following command:
  - UNIX: `nbdevconfig -setconfig -storage_server <storage server host name> -stype PureDisk -configlist <configuration file name>`
  - Windows: `nbdevconfig -setconfig -storage_server <storage server host name> -stype PureDisk -configlist <configuration file name>`

About the MSDP host configuration file

Each NetBackup host that is used for deduplication has a configuration file; the file name matches the name of the storage server, as follows:

storage_server_name.cfg

The *storage_server_name* is the fully qualified domain name if that was used to configure the storage server. For example, if the storage server name is `DedupeServer.example.com`, the configuration file name is `DedupeServer.example.com.cfg`.

The following is the location of the file:

Windows: *install_path*\Veritas\NetBackup\bin\ost-plugins

UNIX: /usr/openv/lib/ost-plugins

See "Configuring media server deduplication in NetBackup" on page 63.

See "Deleting an MSDP host configuration file" on page 200.

See "Changing the MSDP storage server name or storage path" on page 536.

Deleting an MSDP host configuration file

You may need to delete the configuration file from the deduplication hosts. For example, to reconfigure your deduplication environment or disaster recovery may require that you delete the configuration file on the servers on which it exists.

See "About the MSDP host configuration file" on page 199.

See "Changing the MSDP storage server name or storage path" on page 536.

See "Recovering from an MSDP storage server disk failure" on page 583.

To delete the host configuration file

- Delete the file on the deduplication host; its location depends on the operating system type, as follows:

  UNIX: `/usr/openv/lib/ost-plugins`

  Windows: *`install_path`*`\Veritas\NetBackup\bin\ost-plugins`

  The following is an example of the host configuration file name of a server that has a fully qualified domain name:

  ```
  DedupeServer.example.com.cfg
  ```

Resetting the MSDP registry

If you reconfigure your deduplication environment, one of the steps is to reset the deduplication registry.

See "Changing the MSDP storage server name or storage path" on page 536.

Warning: Only follow these procedures if you are reconfiguring your storage server and storage paths.

The procedure differs on UNIX and on Windows.

To reset the MSDP registry file on UNIX and Linux

- Enter the following commands on the storage server to reset the deduplication registry file:

  ```
  rm /etc/pdregistry.cfg
  cp -f /usr/openv/pdde/pdconfigure/cfg/userconfigs/pdregistry.cfg
   /etc/pdregistry.cfg
  ```

To reset the MSDP registry on Windows

- Delete the contents of the following keys in the Windows registry:
  - `HKLM\SOFTWARE\Symantec\PureDisk\Agent\ConfigFilePath`
  - `HKLM\SOFTWARE\Symantec\PureDisk\Agent\EtcPath`

Warning: Editing the Windows registry may cause unforeseen results.

About protecting the MSDP catalog

To increase availability, NetBackup provides a two-tier approach to protect the MSDP catalog, as follows:

Daily shadow copies	NetBackup automatically creates copies of the MSDP catalog. See "About the MSDP shadow catalog" on page 201.
Catalog backup policy	Cohesity provides a utility that you can use to configure a NetBackup policy that backs up the MSDP catalog. See "About the MSDP catalog backup policy" on page 202.

See "About the external MSDP catalog backup" on page 721.

About the MSDP shadow catalog

The NetBackup Deduplication Manager automatically creates a *shadow copy* of the catalog daily. The Deduplication Manager also builds a transaction log for each shadow copy. If NetBackup detects corruption in the MSDP catalog, the Deduplication Manager restores the catalog automatically from the most recent shadow copy. That restore process also plays the transaction log so that the recovered MSDP catalog is current.

By default, the NetBackup Deduplication Manager stores the shadow copies on the same volume as the catalog itself. Cohesity recommends that you store the shadow copies on a different volume.

Warning: You can change the path only during initial MSDP configuration only. If you change it after MSDP backups exist, data loss may occur.

See "Changing the MSDP shadow catalog path" on page 204.

The NetBackup Deduplication Manager creates a shadow copy at 0340 hours daily, host time. To change the schedule, you must change the scheduler definition file.

See "Changing the MSDP shadow catalog schedule" on page 205.

By default, the NetBackup Deduplication Manager keeps five shadow copies of the catalog. You can change the number of copies.

See "Changing the number of MSDP catalog shadow copies" on page 206.

About the MSDP catalog backup policy

Cohesity recommends that you protect the MSDP catalog by backing it up. A NetBackup catalog backup does not include the MSDP catalog. The NetBackup Deduplication Catalog Policy Administration and the Catalog disaster recovery utility (the `drcontrol` utility) configure a backup policy for the MSDP catalog. The policy also includes other important MSDP configuration information.

The MSDP catalog backups provide the second tier of catalog protection. The catalog backups are available if the shadow copies are not available or corrupt.

The following are the attributes for the catalog backup policy that the `drcontrol` utility creates:

Schedule	Weekly **Full Backup** and daily **Differential Incremental Backup**.
Backup window	6:00 A.M. to 6:00 P.M.
Retention	2 weeks

Backup selection

The following are the default catalog paths.

UNIX:

```
/database_path/databases/catalogshadow
/storage_path/etc
/database_path/databases/spa
/storage_path/var
/usr/openv/lib/ost-plugins/pd.conf
/usr/openv/lib/ost-plugins/mtstrm.conf
/database_path/databases/datacheck
```

Windows:

```
database_path\databases\catalogshadow
storage_path\etc
storage_path\var
install_path\Veritas\NetBackup\bin\ost-plugins\pd.conf
install_path\Veritas\NetBackup\bin\ost-plugins\mtstrm.conf
database_path\databases\spa
database_path\databases\datacheck
```

By default, NetBackup uses the same path for the storage and the catalog; the *database_path* and the *storage_path* are the same. If you configure a separate path for the deduplication database, the paths are different. Regardless, the `drcontrol` utility captures the correct paths for the catalog backup selections.

You should consider the following items carefully before you configure an MSDP catalog backup:

- Do not use the **Media Server Deduplication Pool** as the destination for the catalog backups. Recovery of the MSDP catalog from its **Media Server Deduplication Pool** is impossible.
- Use a storage unit that is attached to a NetBackup host other than the MSDP storage server.
- Use a separate MSDP catalog backup policy for each MSDP storage server. The `drcontrol` utility does not verify that the backup selections are the same for multiple storage servers. If the backup policy includes more than one MSDP storage server, the backup selection is the union of the backup selections for each host.
- You cannot use one policy to protect MSDP storage servers on both UNIX hosts and Windows hosts.

UNIX MSDP storage servers require a Standard backup policy and Windows MSDP storage servers require an MS-Windows policy.

See "Configuring an MSDP catalog backup" on page 207.

See "Updating an MSDP catalog backup policy" on page 211.

Changing the MSDP shadow catalog path

You can change the location of the catalog shadow copies. It is recommended that you store the copies on a different volume than both the *storage_path* and the *database_path*. (If you configured a separate path for the deduplication database, the paths are different.)

NetBackup stores the MSDP catalog shadow copies in the following location:

UNIX: */database_path*/databases/catalogshadow

Windows: *database_path*\databases\catalogshadow

Warning: You can change the shadow catalog path during initial MSDP configuration only. If you change it after MSDP backups exist, data loss may occur.

See "About protecting the MSDP catalog" on page 201.

To change the MSDP catalog shadow path

1. Open the following file in a text editor:

   UNIX: */storage_path*/etc/puredisk/spa.cfg

   Windows: *storage_path*\etc\puredisk\spa.cfg

2. Find the CatalogShadowPath parameter and change the value to the wanted path.

   The volume must be mounted and available.

3. After your changes, save the file.

4. Create the .catalog_shadow_identity file in the catalog shadow path that you have specified in step 1.

   Note: There is a period (.) in front of the file name that denotes a hidden file.

   ---

5. Restart the NetBackup Deduplication Manager (spad).

6 Create the shadow catalog directories by invoking the following command on the MSDP storage server:

UNIX: `/usr/openv/pdde/pdcr/bin/cacontrol --catalog backup all`

Windows: *`install_path`*`\Veritas\pdde\cacontrol --catalog backup all`

7 If an MSDP catalog backup policy exists, update the policy with the new shadow catalog directories. To do so, invoke the following command on the MSDP storage server:

UNIX: `/usr/openv/pdde/pdcr/bin/drcontrol --update_policy --policy` *`policy_name`*

Windows: *`install_path`*`\Veritas\pdde\drcontrol --update_policy --policy` *`policy_name`*

Changing the MSDP shadow catalog schedule

NetBackup automatically creates a copy of the MSDP catalog at 0340 hours daily, host time. You can change the default schedule.

See “About protecting the MSDP catalog” on page 201.

To change the MSDP shadow catalog schedule

1. Open the following file in a text editor:

   UNIX: `/database_path/databases/spa/database/scheduler/5`

   Windows: `database_path\databases\spa\database\scheduler\5`

   By default, NetBackup uses the same path for the storage and the catalog; the *database_path* and the *storage_path* are the same. If you configure a separate path for the deduplication database, the paths are different.

   The contents of the file are similar to the following line. The second section of the line (`40 3 * * *`) configures the schedule.

   ```
   CatalogBackup|40 3 * * *|21600|32400|
   ```

2. Edit the second section of the file (`40 3 * * *`). The schedule section conforms to the UNIX `crontab` file convention, as follows:

   ```
   40 3 * * *
   ┬  ┬ ┬ ┬ ┬
   │  │ │ │ │
   │  │ │ │ │
   │  │ │ │ └──── Day of week (0 - 7, Sunday is both 0 and 7, or use
   │  │ │ │       sun, mon, tue, wed, thu, fri, sat; asterisk (*) is
   │  │ │ │       every day)
   │  │ │ └─────────── Month (1 - 12; asterisk (*) is every month)
   │  │ └──────────────── Day of month (1 - 31; asterisk (*) is every
   │  │                   day of the month)
   │  └────────────────────── Hour (0 - 23; asterisk (*) is every hour)
   └──────────────────────────── Minute (0 - 59; asterisk (*) is every
                                 minute of the hour)
   ```

3. After your changes, save the file.
4. Restart the NetBackup Deduplication Manager (`spad`).

Changing the number of MSDP catalog shadow copies

NetBackup keeps five shadow copies of the MSDP catalog. You can change the number of copies.

See "About protecting the MSDP catalog" on page 201.

To change the number of MSDP catalog shadow copies

1. Open the following file in a text editor:

   UNIX: `/storage_path/etc/puredisk/spa.cfg`

   Windows: `storage_path\etc\puredisk\spa.cfg`

2. Find the `CatalogBackupVersions` parameter and change the value to the wanted number of shadow copies. The valid values are 1 to 256, inclusive.
3. After your changes, save the file.
4. Restart the NetBackup Deduplication Manager (`spad`).

Configuring an MSDP catalog backup

Use the following procedure to configure a backup policy for the NetBackup MSDP catalog.

See "About protecting the MSDP data" on page 54.

See "Troubleshooting MSDP catalog backup" on page 786.

To configure an MSDP catalog backup

1. Verify that the MSDP storage server host (that is, the media server) is an additional server for the NetBackup primary server. In the web UI, open the host properties for the media server. Then click **Servers** and click the **Additional servers** tab.

   If the storage server is not in the **Additional servers** list, add the MSDP storage server host to this list. The host *must* be in the **Additional servers** list and *cannot* be in the **Media servers** list.

2. On the MSDP storage server, invoke the `drcontrol` utility and use the appropriate options for your needs. The following is the syntax for the utility:

   Windows: *install_path*\Veritas\pdde\drcontrol--new_policy --residence *residence* [--policy *policy_name*] [--client *host_name*] [--hardware *machine_type*] [--OS *operating_system*] [--dsid *data_selection_ID*] [--NB_install_dir *install_directory*]

   UNIX: /usr/openv/pdde/pdcr/bin/drcontrol--new_policy --residence *residence* [--policy *policy_name*] [--disk_pool *disk_pool_name*] [--client *host_name*] [--hardware *machine_type*] [--OS *operating_system*] [--dsid *data_selection_ID*]

   Descriptions of the options are available in another topic. Note: To ensure that NetBackup activates the policy, you must specify the --residence *residence* option.

   See "MSDP drcontrol options" on page 208.

   The utility creates a log file and displays its path in the command output.

   See "NetBackup MSDP log files" on page 767.

MSDP drcontrol options

The `drcontrol` utility resides in the following directories, depending on host type:

- UNIX: /usr/openv/pdde/pdcr/bin
- Windows: *install_path*\Veritas\pdde

The `drcontrol` utility creates a log file.

See "NetBackup MSDP log files" on page 767.

Table 4-40 describes the options for creating and updating an MSDP catalog backup policy.

Table 4-40 MSDP `drcontrol` options for catalog backup and recovery

Option	Description
`--auto_recover_DR`	Recover the MSDP catalog from the most recent backup image. This option automatically recovers the catalog and performs all of the actions necessary to return MSDP to full functionality. This option requires the `--policy` *`policy_name`* option. To recover the catalog from a backup other than the most recent, contact your Cohesity Support representative.
`--client` *`host_name`*	The client to back up (that is, the host name of the MSDP storage server). Default: the value that `bpgetconfig CLIENT_NAME` returns.
`--cleanup`	Remove all of the old MSDP catalog directories during the catalog recovery process. Those directories are renamed during the recovery.
`--disk_pool`	This option is required for `auto_recover_DR` when the disk pool name cannot be determined from the host name.
`--dsid`	The data selection ID is the catalog directory for one of the NetBackup domains. In a multi-domain scenario when you recover the catalog from another domain, the `dsid` of the other NetBackup domain is used. To obtain the `dsid` of the other NetBackup domain, run the `spauser` command to list the `dsid`. The default value is 2.
`--hardware` *`machine_type`*	The hardware type or the computer type for the host. Spaces are not allowed. If the string contains special characters, enclose it in double quotation marks ("). Default: Unknown.
`--initialize_DR`	Performs the following actions to prepare for MSDP catalog recovery: ■ Verifies that the most recent catalog backup is valid. ■ Stops the deduplication services. ■ Moves the existing catalog files so that they are empty for the recovery.
`--list_files`	List the files in the most recent MSDP catalog backup. This option requires the `--policy` *`policy_name`* option.

Table 4-40 MSDP `drcontrol` options for catalog backup and recovery *(continued)*

Option	Description
`--log_file` *pathname*	The pathname for the log file that the `drcontrol` utility creates. By default, the utility writes log files to */storage_path*`/log/drcontrol/`.
`--NB_install_dir` *install_directory*	Windows only. Required option if NetBackup was installed in a location other than the default (`C:\Program Files\Veritas`). If the string contains spaces or special characters, enclose it in double quotation marks ("). Do not use a trailing backslash in the *install_directory* string.
`--new_policy`	Create a new policy to protect the deduplication catalog on this host. If a policy with the given name exists already, the command fails. **Note:** To ensure that NetBackup activates the policy, you must specify the `--residence` *residence* option.
`--OS` *operating_system*	The operating system for the host. Spaces are not allowed. If the string contains special characters, enclose it in double quotation marks ("). Default: `UNIX/Linux` or `MS-Windows`.
`--policy` *policy_name*	The name for the backup policy. Required with `--auto_recover_DR` and `--update_policy`; optional with `--new_policy`. Default: `Dedupe_Catalog_`*shorthostname*
`--print_space_required`	Display an estimate of the percentage of file system space that is required to recover the MSDP catalog.
`--recover_last_image`	Restore the MSDP catalog from the last set of backup images (that is, the last full plus all subsequent incrementals). The `drcontrol` utility calls the NetBackup `bprestore` command for the restore operation.
`--refresh_shadow_catalog`	Deletes all existing shadow catalog copies and creates a new catalog shadow copy.

Table 4-40 MSDP `drcontrol` options for catalog backup and recovery *(continued)*

Option	Description
`--residence` *residence*	The name of the storage unit on which to store the MSDP catalog backups. Do not use the **Media Server Deduplication Pool** as the destination for the catalog backups. Recovery of the MSDP catalog from its **Media Server Deduplication Pool** is impossible. Cohesity recommends that you use a storage unit that is attached to a NetBackup host other than the MSDP storage server.
`--update_policy`	Update a policy, as follows: ■ If the client name (of this media server) is not in the policy's client list, add the client name to the policy's client list. ■ If you specify the `--OS` or `--hardware` options, replace the values currently in the policy with the new values. ■ Update the backup selection based on the locations of the MSDP storage directories and configuration files. Therefore, if you modify any of the following, you must use this option to update the catalog backup policy: ■ Any of the following values in the `spa.cfg` file (`section:variable` pairs): ■ `StorageDatabase:CatalogShadowPath` ■ `StorageDatabase:Path` ■ `Paths:Var` ■ The `spa.cfg` or `contentrouter.cfg` locations in the `pdregistry.cfg` file. This option fails if there is no policy with the given policy name. It also fails if the existing policy type is incompatible with the operating system of the host on which you run the command. This option requires the `--policy` *policy_name* option.
`--verbose`	Echo all `drcontrol` log statements to stdout.

See "Configuring an MSDP catalog backup" on page 207.

Updating an MSDP catalog backup policy

You can use any NetBackup method to update an MSDP catalog backup policy manually. However, you should use the NetBackup Deduplication Catalog Policy Administration and Catalog Disaster Recovery (`drcontrol`) under the following circumstances:

- To add the client name of the storage server to the policy's client list.
- To update the `--OS` value.
- To update the `--hardware` value.
- To update the backup selection if you modified any of the following configuration values:
  - Any of the following values in the `spa.cfg` file (`section:variable` pairs):
    - `StorageDatabase:CatalogShadowPath`
    - `StorageDatabase:Path`
    - `Paths:Var`
  - The `spa.cfg` or `contentrouter.cfg` locations in the `pdregistry.cfg` file.

See "About protecting the MSDP data" on page 54.

See "Troubleshooting MSDP catalog backup" on page 786.

To update an MSDP catalog backup

- On the MSDP storage server, invoke the `drcontrol` utility and use the appropriate options for your needs. The following is the syntax for an update operation:

  UNIX: `/usr/openv/pdde/pdcr/bin/drcontrol--update_policy --policy` *`policy_name`* `[--client` *`host_name`*`] [--hardware` *`machine_type`*`] [--OS` *`operating_system`*`]`

  Windows: *`install_path`*`\Veritas\pdde\drcontrol--update_policy --policy` *`policy_name`* `[--client` *`host_name`*`] [--hardware` *`machine_type`*`] [--OS` *`operating_system`*`] [--OS` *`operating_system`*`] [--NB_install_dir` *`install_directory`*`]`

  Descriptions of the options are available in another topic.

  See "MSDP drcontrol options" on page 208.

  The utility creates a log file and displays its path in the command output.

  See "NetBackup MSDP log files" on page 767.

About MSDP FIPS compliance

The Federal Information Processing Standards (FIPS) define U.S. and Canadian Government security and interoperability requirements for computer systems. The FIPS 140-2 standard specifies the security requirements for cryptographic modules.

It describes the approved security functions for symmetric and asymmetric key encryption, message authentication, and hashing.

For more information about the FIPS 140-2 standard and its validation program, see the National Institute of Standards and Technology (NIST) and the Communications Security Establishment Canada (CSEC) Cryptographic Module Validation Program website at https://csrc.nist.gov/projects/cryptographic-module-validation-program.

The NetBackup MSDP is now FIPS validated and can be operated in FIPS mode.

Note: You must run FIPS mode on a new installation of NetBackup 8.1.1. You can only enable OCSD FIPS on NetBackup 10.0 and newer versions.

Enabling MSDP FIPS mode

Ensure that you configure the storage server before you enable the MSDP FIPS mode.

Caution: Enabling MSDP FIPS mode might affect the NetBackup performance on a server with the Solaris operating system.

Enable the FIPS mode for MSDP by running the following commands:

- For UNIX:
  `/usr/openv/pdde/pdag/scripts/set_fips_mode.sh 1`
  For Windows:
  `<install_path>\Veritas\pdde\set_fips_mode.bat 1`
- Restart the NetBackup service on the server and the client.
  - For UNIX:
    - `/usr/openv/netbackup/bin/bp.kill_all`
    - `/usr/openv/netbackup/bin/bp.start_all`
  - For Windows:
    - `<install_path>\NetBackup\bin\bpdown`
    - `<install_path>\NetBackup\bin\bpup`

Enable the FIPS mode for MSDP or OpenCloudStorageDaemon (OCSD) by performing the following:

- Use existing tool to enable or disable OCSD FIPS. Using this method changes the entire MSDP FIPS configuration.

- For Windows:

  `<install_path>\Veritas\pdde\set_fips_mode.bat 1`

- For UNIX:

  `/usr/openv/pdde/pdag/scripts/set_fips_mode.sh 1`

- In NetBackup, `OCSD` FIPS is disabled by default. Enable or disable `OCSD` FIPS by changing the `OpenCloudStorageDaemon/FIPS`:

  `/etc/pdregistry.cfg`

Restart the NetBackup services on the server and the client for these changes to take effect:

- For Windows:
  - `<install_path>\NetBackup\bin\bpdown`
  - `<install_path>\NetBackup\bin\bpup`
- For UNIX:
  - `/usr/openv/netbackup/bin/bp.kill_all`
  - `/usr/openv/netbackup/bin/bp.start_all`

Warning: For security reasons, the recommendation is that you do not disable the MSDP FIPS mode once it has been enabled.

Getting the status of MSDP FIPS mode

To get status of the MSDP FIPS mode, enter the following commands:

For UNIX:

```
/usr/openv/pdde/pdcr/bin/crcontrol --getmode
```

For Windows:

```
<install_path>\Veritas\pdde\crcontrol.exe --getmode
```

Other things to note:

- FIPS must be enabled on all the NetBackup components to establish a connection. When the FIPS mode is not enabled, communication can occur between the NetBackup clients and the servers that have earlier, supported NetBackup versions.

Configuring the NetBackup client-side deduplication to support multiple interfaces of MSDP

If you have network configurations like VLAN or Subnet with NetBackup client, then the MSDP server has multiple network interfaces. These interfaces are connected to different switches or VLANs. As MSDP has only one storage server, NetBackup clients cannot access the MSDP server using the storage server name. The deduplication can fail on the clients.

You can add support for up to 30 interfaces.

Run the following steps to use the `cacontrol` command option (Location: `/usr/openv/pdde/pdcr/bin/`) to configure MSDP and specify the network interfaces that the NetBackup client can use:

1. Log on the MSDP server.
2. Use the following command to add the alternate interfaces:

   ```
   cacontrol --interface add msdp-a.server.com
   ```

   You can remove an added interface using the following command:

   ```
   cacontrol --interface remove msdp-a.server.com
   ```

3. Use either of the following options to validate the interface configuration:
   - `cacontrol --interface list`
   - `bpstsinfo -si -storage_server msdp-a.server.com -stype PureDisk`
     Location of the `bpstsinfo` command: `/usr/openv/netbackup/bin/admincmd/`
4. Configure the NetBackup client-side deduplication backup policy and run the backup operation.

About MSDP multi-domain support

An MSDP storage server is configured in a NetBackup media server. The NetBackup media servers and clients in the NetBackup domain can use this storage server. By default, the NetBackup media servers and clients cannot directly use an MSDP storage server from another NetBackup domain. For example, NetBackup media servers or clients cannot backup data to an MSDP storage server from another NetBackup domain.

To use an MSDP storage server from another NetBackup domain, the MSDP storage server must have multiple MSDP users. Then NetBackup media servers or clients can use the MSDP storage server from another NetBackup domain by using a different MSDP user. Multiple NetBackup domains can use the same MSDP storage server, but each NetBackup domain must use a different MSDP user to access that MSDP storage server.

To add an MSDP user on an MSDP storage server, run the following command:

- Windows

  ```
  <install_path>\pdde\spauser -a -u <username> -p <password> --role
  admin
  ```

- UNIX

  ```
  /usr/openv/pdde/pdcr/bin/spauser -a -u <username> -p <password>
  --role admin
  ```

If the storage server is NetBackup WORM Storage Server or NetBackup Flex Scale storage server, run following NetBackup Deduplication Shell command:

```
setting MSDP-user add-MSDP-user username=<username> [role=<role-name>]
```

Where, `role` is optional, which can be `admin` or `app`. If the role is not specified, the default role `admin` is used.

To list all the MSDP users, run the following command on the MSDP storage server:

- Windows

  ```
  <install_path>\pdde\spauser -l
  ```

- UNIX

  ```
  /usr/openv/pdde/pdcr/bin/spauser -l
  ```

If the storage server is NetBackup WORM Storage Server or NetBackup Flex Scale storage server, run following NetBackup Deduplication Shell command:

```
setting MSDP-user list
```

Note: We recommend that the total number of MSDP users that you create to support the multi-domain should not exceed 128 users.

To use an MSDP storage server from another NetBackup domain, you must obtain a NetBackup certificate from another NetBackup domain.

Run the following commands on every NetBackup media server or client that wants to use an MSDP storage server from another domain:

- Windows

```
install_path\NetBackup\bin\nbcertcmd -getCACertificate -server
another_primary_server
install_path\NetBackup\bin\nbcertcmd -getCertificate -server
another_primary_server -token token_string
```

- UNIX

```
/usr/openv/netbackup/bin/nbcertcmd -getCACertificate -server
another_primary_server
/usr/openv/netbackup/bin/nbcertcmd -getCertificate -server
another_primary_server -token token_string
```

If the storage server is NetBackup WORM Storage Server or NetBackup Flex Scale storage server, run the following NetBackup Deduplication Shell command:

```
setting certificate get-CA-certificate
primary_server=another_primary_server

setting certificate get-certificate
primary_server=another_primary_server token=token_string
```

Use either of these two methods to obtain the authorization tokens:

- NetBackup web UI
  - On the left, click **Security > Tokens**.
  - Click **Add** to create a token.
- NetBackup commands
  - Use the `bpnbat` command to log on the target NetBackup primary server.
  - Use the `nbcertcmd` command to get the authorization tokens.
    For more information on the commands, refer to the *NetBackup Commands Reference Guide*.

An example for using an MSDP storage server from another NetBackup domain

The following table describes the hierarchy that is used in the example:

NetBackup domain A	NetBackup domain B
primaryA	primaryB
mediaA1	mediaB
mediaA2	
clientA	

PrimaryA is the host name of the primary server of NetBackup domain A and the domain contains two media servers (mediaA1, mediaA2), and one client (clientA). PrimaryB is the host name of the primary server of NetBackup domain B and the domain contains one media server (mediaB).

Using the following sample steps, create an MSDP storage server in domain B and let domain A use the MSDP storage server:

1. Create an MSDP storage server on the media server mediaB of NetBackup domain B.
   - Open the web UI.
   - On the left, click **Storage > Disk storage**.
   - On the **Storage servers** tab, click **Add** and select **Media Server Deduplication Pool to local or cloud storage**.
2. Run the following command on mediaB to create a new MSDP user testuser1 with password as testuser1pass.

   ```
   spauser -a -u "testuser1" -p "testuser1pass" --role admin
   ```
3. Run the following command on mediaA1 to get a CA certificate and a host certificate from primaryB.

   ```
   nbcertcmd -GetCACertificate -server primaryB
   ```

   ```
   nbcertcmd -GetCertificate -server primaryB -token <token_string>
   ```
4. Create an MSDP OpenStorage server on mediaA1 of NetBackup domain A.
   - Open the web UI.
   - On the left, click **Storage > Disk storage**.
   - On the **Storage servers** tab, click **Add** and select **OpenStorage Technology**.

   Then the OpenStorage server type is **PureDisk**, the Storage server name is mediaB, the username is testuser1, and the password is testuser1pass.

   You must enter the server type as **PureDisk**.

   Now mediaA1 of the NetBackup domain can use the MSDP storage server mediaB. To use mediaA2 as a load balance server of the MSDP storage server, you can run the following certificate command on mediaA2:
   - `nbcertcmd -GetCACertificate -server primaryB`
   - `nbcertcmd -GetCertificate -server primaryB -token <token_string>`

To run client-direct backup from `clientA` to MSDP storage server `mediaB`, run the following certificate command on `clientA`:

- `nbcertcmd -GetCACertificate -server primaryB`
- `nbcertcmd -GetCertificate -server primaryB -token <token_string>`

5. After creating the MSDP OpenStorage server, create a related NetBackup disk pool and storage unit. Use the storage unit to run all the related NetBackup jobs.

When multi-domain is used with optimized duplication or A.I.R., there is communication between the MSDP storage servers from two different NetBackup domains. The MSDP storage server from the other domain must have a certificate generated by primary server of the local NetBackup domain. Run the `nbcertcmd` commands on the source side MSDP storage server to request a certificate from the NetBackup primary server of the target MSDP storage server.

When backup and restore jobs on the client and multi-domain are used together, there is communication between the NetBackup client and MSDP storage server from two different NetBackup domains. Run the `nbcertcmd` commands on the NetBackup client to request a certificate from the NetBackup primary server of MSDP storage server.

When one NetBackup domain uses the MSDP storage server of another NetBackup domain, the MSDP storage server cannot be the A.I.R target of that NetBackup domain.

If an external CA is used in the NetBackup setup, you do not need to run the `nbcertcmd –GetCACertificate` and the `nbcertcmd –GetCertificate` commands. If NetBackup domains A and B do not use the same external CA, synchronize the external root CA between the two NetBackup domains for MSDP communication.

For more information of the external CA, refer to *NetBackup Security and Encryption Guide*.

When one NetBackup domain uses an MSDP storage server that has multiple network interfaces and related host names, another NetBackup domain can use any one host name to configure the OpenStorage server. If the MSDP storage server that has multiple host names uses an external CA, the **Subject Alternative Name** field of the external certificate must contain all the host names that are used to configure the OpenStorage server.

Only local storage of one MSDP storage can be used by other NetBackup domains. Cloud LSU in one MSDP storage server cannot be used by other NetBackup domains. Different NetBackup domains should not use the same MSDP user to access the MSDP storage server or it will cause the storage server to shut down

after several minutes. To solve this issue, See “Trouble shooting multi-domain issues” on page 791.

If there is a data corruption in the MSDP storage server, only the first domain of the MSDP storage server receives the data corruption notification. With multi-domain, one storage server is used by multiple NetBackup domains, and each domain can check space usage of this storage server. The used space of the storage server is the sum of data from all domains.

Note: The universal share backup to the target domain is not supported in the multi-domain setup.

About MSDP application user support

You can create an MSDP application user specifically to work with NetBackup Dedupe Direct for Oracle. NetBackup Dedupe Direct for Oracle is a lightweight plug-in that you can use to store the data from RMAN backups to MSDP storage directly.

For more information about NetBackup Dedupe Direct for Oracle, see *NetBackup for Oracle Administrator's Guide*.

Use the **spauser** command-line tool on MSDP server to manage MSDP application users.

To manage the MSDP application users

1. Log on the MSDP server.
2. Create an application user.

```
/usr/openv/pdde/pdcr/bin/spauser -a -u <username> -p <password>
--role app
```

3. Delete an application user.

```
/usr/openv/pdde/pdcr/bin/spauser -d -u username [-p password]
```

4. Change an application user password.

```
/usr/openv/pdde/pdcr/bin/spauser -c -u username [-p oldpassword
-q newpassword]
```

5. List all users.

```
/usr/openv/pdde/pdcr/bin/spauser -l
```

About MSDP mutli-domain VLAN Support

MSDP supports multi-domain NetBackup setups. In a multi-domain set-up, it is important for primary servers from other domains to connect with the MSDP storage server and the primary server of the NetBackup domain that contains the MSDP server. The primary servers and media servers must have multiple network interfaces and host names in a multi-domain setup.

When you configure MSDP VLAN, the local NetBackup domain and the other NetBackup domain must have the NetBackup version 8.2 or later.

An example for using an MSDP VLAN

The following table describes the hierarchy that is used in the example:

NetBackup domain A	NetBackup domain B
`primaryA - (10.XX.30.1/24)`	`primaryB - (10.XX.40.3/24)`
`primaryA2 - (10.XX.40.1/24)`	`mediaB - (10.XX.40.4/24)`
`mediaA - (10.XX.30.2/24)`	
`mediaA2 - (10.XX.40.2/24)`	

`primaryA` is the primary server of domain A and has two host names and IP addresses. `mediaA` is the media server of domain A and has two host names and IP addresses. MSDP storage server is created on media server `mediaA`.

To let domain B access the MSDP storage server on `mediaA` of domain A, run the following steps:

1. Create an MSDP storage server on media server `mediaA` of NetBackup domain A.

   Open the NetBackup web UI. Click **Storage > Disk storage**. Click on the **Storage servers** tab. Click **Add** and select **Media Server Deduplication Pool to local or cloud storage**.

2. Run following command on `mediaA` to create a new MSDP user `testuser1` with password `testuser1pass`:

   ```
   spauser -a -u "testuser1" -p "testuser1pass" --role admin
   ```

3. Servers in the domain B can only access IP like 10.XX.40.*, so `primaryA2` is used as the primary server host name of domain A.

   Run following command on `mediaB` to get a CA certificate and a host certificate from `primaryA`:

   ```
   nbcertcmd -GetCACertificate -server primaryA2
   ```

```
nbcertcmd -GetCertificate -server primaryA2 -token <token_string>
```

If the `nbcertcmd -GetCACertificate` displays the error "The server name does not match any of the host names listed in the server's certificate", refer to the following article to add more host name to primary server:

https://support.cohesity.com/s/article/article-100034092

4. Create an MSDP OpenStorage server on `mediaB` of NetBackup domain B.

   Open the NetBackup web UI. Click **Storage > Disk storage**. Click on the **Storage server** tab. Click **Add** and select **OpenStorage Technology (OST)**.

   The OpenStorage server name `mediaA2` is used as the host name that has the IP address 10.XX.40.*.

   OpenStorage server type is **PureDisk**, user name is `testuser1`, and password is `testuser1pass`. You must enter the server type as **PureDisk**.

Now `mediaB` of NetBackup domain B can use the MSDP storage server `mediaA2` and the network IP address 10.XX.40.*

In a multi-domain NetBackup configuration, there are times when the media server in domain B must know the server certificate of the media server in domain A. For example, this setup is required when a VMware image recovery is performed from domain A to domain B.

To move `mediaA2`'s server certificate to `primaryB` is a two-step process, and you need to be a privileged user to run the following steps:

1. Read the file `/etc/nginx/conf.d/nginx_spws.conf` and examine the `ssl_certificate` field.

   Copy `ssl_certificate` field (for example, `/msdp/data/dp1/pdvol/spws/var/keys/cluster_spws_cert.pem`) from mediaA2 to primaryB.

   Sample file:

```
engine : cat /etc/nginx/conf.d/nginx_spws.conf
server {
listen 443 default ssl;
listen [::]:443 default ssl;
server_name _;
ssl_session_cache shared:SSL:1m;
ssl_session_timeout 5m;
ssl_certificate
/msdp/data/dp1/pdvol/spws/var/keys/cluster_spws_cert.pem;
ssl_certificate_key
/msdp/data/dp1/pdvol/spws/var/keys/cluster_spws_key.pem;
```

```
ssl_verify_client off;
ssl_protocols TLSv1.2;
ssl_ciphers
ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-SHA256:ECDHE-RSA-AES256-G
CM-SHA384:ECDHE-RSA-AES256-SHA384:DHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES
128-SHA256:DHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES256-SHA256;
ssl_prefer_server_ciphers on;
include /etc/nginx/locations/nginx_loc_spws.conf;
}
```

2. Run the following command on `primaryB` to import that certificate to `primaryB`'s trusted keystore.

Note: The `storepass` and `keypass` values can be found in `/usr/openv/var/global/jkskey` on `primaryB`.

```
/usr/openv/java/jre/bin/keytool
-keystore
/usr/openv/var/global/wsl/credentials/truststoreMSDP.bcfks
-storetype BCFKS
-providername CCJ
-providerclass
com.safelogic.cryptocomply.jcajce.provider.CryptoComplyFipsProvider
-providerpath /usr/openv/wmc/webserver/lib/ccj.jar
-storepass
4d2b912d38dff8d2406b2aba2d023740aafa520a16cd3bb8b1b39b10b58a4ce5
-keypass
4d2b912d38dff8d2406b2aba2d023740aafa520a16cd3bb8b1b39b10b58a4ce5
-alias primaryB -importcert -file cluster_spws_cert.pem
```

Note: Your version of the `bc-fips-X.X.X.X.jar` file can be different than the one in the previous example. Search that directory for `bc-fips*` to find the right version for your NetBackup installation.

---

3. When you run the `-list` command on `primaryB`, you should see something similar to the following example:

```
/usr/openv/java/jre/bin/keytool -list
-keystore
/usr/openv/var/global/wsl/credentials/truststoreMSDP.bcfks
-storetype BCFKS
```

```
-providername CCJ
-providerclass
com.safelogic.cryptocomply.jcajce.provider.CryptoComplyFipsProvider
-providerpath /usr/openv/wmc/webserver/lib/ccj.jar
-storepass
4d2b912d38dff8d2406b2aba2d023740aafa520a16cd3bb8b1b39b10b58a4ce5

Keystore type: BCFKS
Keystore provider: BCFIPS
Your keystore contains 2 entries
ca1, Jan 11, 2023, trustedCertEntry,
Certificate fingerprint (SHA-256):
4A:52:C8:9E:B1:1F:A9:21:99:3B:AA:5A:0C:B5:C3:2F:51:(string
continues)
primaryB, Jan 16, 2023, trustedCertEntry,
Certificate fingerprint (SHA-256):
AE:34:D1:63:B1:94:33:8C:07:5D:9A:D6:2B:CF:5B:52:D7:(string
continues)
```

The second certificate that is listed is that of `mediaA2`.

If an external CA is used in the NetBackup setup, you do not need to run the `nbcertcmd -GetCACertificate` and the `nbcertcmd -GetCertificate` commands. If NetBackup domain A and NetBackup domain B do not use the same external CA, you must synchronize the external root CA between the two NetBackup domains for MSDP communication. If the servers have multiple host names, then the **Subject Alternative Name** field of the external certificate must contain all the host names.

About NetBackup WORM storage support for immutable and indelible data

NetBackup WORM storage server supports immutable and indelible data storage.

For more information, refer to the *Configuring immutability and indelibility of data in NetBackup* chapter in the *NetBackup Administrator's Guide, Volume I*.

NetBackup WORM storage and retention period

A retention period lets you define a time for protecting the backup image. Once you define a retention period, MSDP stores a timestamp along with the image metadata to indicate when the retention period expires. After the retention period expires, the image data can be deleted.

You can set the following parameters for the retention period:

- Lock Minimum Duration
- Lock Maximum Duration

For more information refer to the *Workflow to configure immutable and indelible data* topic in the *NetBackup Administrator's Guide, Volume I*.

WORM storage supports the following retention period modes:

- Compliance mode
  Any type of user cannot overwrite or delete the data that is protected using the compliance mode for the defined retention period. Once you set a retention period for the data storage, you cannot shorten it and can only extend it.
- Enterprise mode
  Users require special permissions to disable the retention lock and then delete the image. Only the MSDP security administrator user can disable the retention lock and then the delete the image if required. You can use the enterprise mode to test the retention period behavior before you create a compliance mode retention period.

See "About the NetBackup command line options to configure immutable and indelible data" on page 225.

See "About the NetBackup Deduplication Shell" on page 709.

About the NetBackup command line options to configure immutable and indelible data

As a security administrator, you can use the following `catdbutil` and `spadb` command line options to configure immutable and indelible data or WORM storage.

See "About NetBackup WORM storage support for immutable and indelible data " on page 224.

The `catdbutil` command lets you query and modify catalog database. The command is available at the following location:

```
/usr/openv/pdde/pdcr/bin/
```

The following table describes the WORM-specific options and arguments for the `catdbutil` command.

Table 4-41 The options and arguments for the `catdbutil` command.

Command and its description	Option	Description
`catdbutil` Query and modify catalog database.	`worm list` Usage: `--worm list [--pattern PATTERN]`	Display the backup IDs and other information of the WORM-enabled images. The following information is displayed: `backupid`, `retention lock date`, `time left`, `worm flags`
	`worm disable` Usage: --worm disable --backupid	Disable retention lock for an image using the backup ID.
	`worm audit` Usage: --worm audit [--sdate *yyyy-MM-ddThh:mm:ss* | --edate *yyyy-MM-ddThh:mm:ss*]	Display WORM audit information for a specified date and time interval.

The `spadb` command line utility that lets you use the NetBackup Deduplication Manager (`spad`) to set WORM for an LSU and define the WORM mode and the interval for making the image immutable and indelible.

The Deduplication Manager reads the WORM mode from the `/etc/lockdown-mode.conf file` file.

The command is available at the following location:

`/usr/openv/pdde/pdcr/bin/`

The following table describes the WORM-specific options and arguments for the `spadb` command.

Table 4-42 The options and arguments for the `spadb` command.

Command and its description	Option	Description
`spadb` Command line utility that lets you use the NetBackup Deduplication Manager (`spad`)	`spadb update WORM set ${FIELD1_NAME}=`**`xxx`**`, ${FIELD2_NAME}=`*`xxxx`*` where id=${`*`DSID`*`} #` `field names:` ■ `indeliable_minimum_interval` ■ `indeliable_maximum_interval`	Use the data selection ID to configure the following WORM properties: ■ `indelible_minimum_interval` and `indelible_maximum_interval` Set the minimum and maximum interval in days for making the image indelible. For example, `spadb -c "update WORM set indelible_minimum_interval=1 where dsid=2"` `spadb -c "update WORM set indelible_maximum_interval=1000000 where dsid=2"`

Running MSDP services with the non-root user

You can reduce the security risks by running the applications with the non-root user.

Starting from NetBackup 10.3, you can run MSDP daemons with the non-root user on NetBackup BYO (Red Hat Enterprise Linux and SUSE), NetBackup Appliance, and NetBackup media server on Flex Appliance.

The non-root user is the MSDP service user that is configured automatically in MSDP configuration file `/etc/pdregistry.cfg` after MSDP service user is changed. You are recommended to use the non-root user as the MSDP service user.

Changing the service user after installation or upgrade

After NetBackup is installed, you create MSDP storage server. If NetBackup is configured to run with the non-root service user before MSDP storage server is created, MSDP can also be run with the service user automatically.

If MSDP services are not running as the service user, you can manually change the service user by using the `msdpserviceusercmd` command.

Following are the prerequisites to run MSDP with the non-root service user:

- Check if NetBackup services can be run with the non-root user. If NetBackup services cannot be run with the non-root user, change the NetBackup service user account. For more information, see the *NetBackup Security and Encryption Guide*.

- On the NetBackup BYO, run the following command to check the maximum number of files that service user can open:

  `ulimit -Hn`

  Set the limit to `1048576` in `/etc/security/limits.conf` file.

To change the MSDP service user on NetBackup BYO

1. Stop the following services:

   ```
   systemctl stop crond.service

   /usr/openv/netbackup/bin/bp.kill_all

   /opt/VRTSpbx/bin/vxpbx_exchanged stop
   ```

2. Run the following command to change the MSDP service user.

   ```
   /usr/openv/pdde/pdconfigure/scripts/support/msdpserviceusercmd
   ```

3. Start the following services:

   ```
   /opt/VRTSpbx/bin/vxpbx_exchanged start

   /usr/openv/netbackup/bin/bp.start_all

   systemctl start crond.service
   ```

To change the MSDP service user on the media server on Flex Appliance

1. Stop the following services:

   ```
   /opt/veritas/vxapp-manage/health disable

   systemctl stop crond.service

   /opt/veritas/vxapp-manage/stop
   ```

2. Run the following command to change the MSDP service user.

   ```
   /usr/openv/pdde/pdconfigure/scripts/support/msdpserviceusercmd
   ```

3. Start the following services:

   ```
   /opt/veritas/vxapp-manage/start

   systemctl start crond.service

   /opt/veritas/vxapp-manage/health enable
   ```

To change the MSDP service user on NetBackup Appliance

1. Stop the `crond` service from the NetBackup Appliance shell menu.

   ```
   Main_Menu > Support > Service Stop crond
   ```

   For the usage of NetBackup Appliance Shell Menu, see the *NetBackup Appliance Commands Reference Guide*.

2. Stop the NetBackup processes from the NetBackup Appliance shell menu.

   ```
   Main_Menu > Support > Processes > NetBackup Stop
   ```

3. Run the following command from the NetBackup CLI to change the MSDP service user.

   ```
   nbucliuser-!> msdpserviceusercmd
   ```

   For the usage of NetBackup CLI, see *About the NetBackupCLI user role* topic of the *NetBackup Appliance Security Guide*.

4. Start NetBackup processes from the NetBackup Appliance shell menu.

   ```
   Main_Menu > Support > Processes > NetBackup Start
   ```

5. Start the `crond` service from the NetBackup Appliance shell menu.

   ```
   Main_Menu > Support > Service Restart crond
   ```

Note: MSDP service user is same as NetBackup service user.

`msdpserviceusercmd` can take long time depending on the MSDP storage data size. If you think that the command may be interrupted (for example, you turn off the laptop), run `msdpserviceusercmd` command in the background using Linux command `nohup`.

If `msdpserviceusercmd` is interrupted, MSDP service fails to start. In that case, run the command again to restart the process to change the service user.

When you add an additional MSDP storage volume using the command `crcontrol --dsaddpartition [volume path]`, ensure that the MSDP service user has the read and write permissions on the new storage volume path.

The services `spad`, `spoold`, `ocsd`, and `s3srv` are the MSDP services that run with the service user. MSDP web service `spws` always runs with the `spws` user.

Running MSDP commands with the non-root user

When an MSDP command is run by the root user, it automatically switches to the service user. However, if a non-root user needs to run MSDP commands, they can

use the `msdpcmdrun` wrapper command. This tool is supported on NetBackup BYO, media servers on Flex Appliance, NetBackup CloudScale, and NetBackup Appliances.

If `nbcmdrun` is configured and enabled, you can use `msdpcmdrun` as follows:

```
/usr/openv/netbackup/bin/nbcmdrun msdpcmdrun <msdp commands>
```

For more information about `nbcmdrun`, see *Running NetBackup commands using the nbcmdrun wrapper command* topic of the *NetBackup Security and Encryption Guide*.

Note: The `nbcmdrun` command does not support passing environment variables or user inputs to MSDP commands.

Alternatively, you can run `msdpcmdrun` directly without `nbcmdrun` as follows:

```
/usr/openv/pdde/pdcr/bin/msdpcmdrun <msdp_commands>
```

See "Troubleshooting the msdpcmdrun issues" on page 794.

Configuring a non-root user to run MSDP commands through msdpcmdrun

On a server where the MSDP storage server is configured, a non-root user can also run MSDP commands through a wrapper called `msdpcmdrun`.

Every command run through `msdpcmdrun` is audited. Audit report is displayed in the NetBackup web UI under **Security > Security events > Audit events**.

Prerequisites:

- Only the users from the deduplication administrator group **pdadmin** can run MSDP commands through `msdpcmdrun`.
- Admin user must create a deduplication administrator group **pdadmin** if it does not exist and add required users to the user group.
- If the MSDP storage server was already configured before creating the **pdadmin** group, the admin user must start the PureDisk Privilege Access Service (PDPAS) service manually.

To configure and enable msdpcmdrun for a non-root user:

1. Create the **pdadmin** group if it does not exist already.

   ```
   groupadd pdadmin
   ```

2. Add the user to the **pdadmin** group. Either create a new user or add an existing user to the **pdadmin** group.

   ```
   useradd test -G pdadmin
   ```

3. Ensure that the PDPAS service is running.

   - Check if the PDPAS service is active.

     ```
     /usr/openv/pdde/pdconfigure/pdde ps | grep pdpas
     ```

   - If the service is not running, start the service.

     ```
     /usr/openv/pdde/pdconfigure/pdde pdpas start
     ```

4. Run `msdpcmdrun` as a non-root user.

   For example, `/usr/openv/pdde/pdcr/bin/msdpcmdrun crstats`

See "Troubleshooting the msdpcmdrun issues" on page 794.

About PDPAS logging

The PDPAS log can be found in the `/<MSDP storage>/logs/pdpas.log` file. By default, the log level is set to 1, which logs information, warning, and error messages.

To set the desired log level, modify the `loglevel` parameter under the PDPAS section of the `/etc/pdregistry.cfg` file. The value ranges from 1 to 4.

For example,

```
[PDPAS]
loglevel=1
logpath=/<storage-server-dir>/log
```

Table 4-43 PDPAS log levels

Log level	Description
1	Logs information, warning, and error messages.
2	Logs warning and error messages.
3	Logs only error messages.
4	Logs information, warning, error, and debug/trace messages. This is the highest level of logging.

Sample msdpcmdrun commands

Run the following command to list the supported MSDP commands.

```
/usr/openv/pdde/pdcr/bin/msdpcmdrun -l
```

Table 4-44 Sample `msdpcmdrun` commands

Task	Command
Get MSDP LSU data statistics.	`/usr/openv/pdde/pdcr/bin/msdpcmdrun crstats`
List all users.	`/usr/openv/pdde/pdcr/bin/msdpcmdrun spauser -l`
List the immutable cloud volumes and configurations. It requires environment variables to be set.	`export MSDPC_ACCESS_KEY=AccessKeyID` `export MSDPC_SECRET_KEY=SecretAccessKey` `export MSDPC_REGION=us-east-1` `export MSDPC_PROVIDER=amazon` `/usr/openv/pdde/pdcr/bin/msdpcmdrun msdpcldutil list`

Considerations for file access

When running MSDP commands as the service user, if the command requires a file path, ensure that the file is accessible to the service user.

For example, `msdpcldutil list --credfile /tmp/env.txt`

In this case, a file `/tmp/env.txt` should be readable by the MSDP service user, which can be found in the `/etc/pdregistry.cfg` file under the `MSDP_SERVICE_USER` configuration.

Running the MSDP commands on NetBackup appliances

On a NetBackup Appliance, log in to the appliance shell menu using the NetBackup CLI user. From there, you can run `msdpcmdrun` commands directly in the shell:

```
nbcliuser-!> msdpcmdrun catdbutil --count
```

For the usage of NetBackup CLI, see *About the NetBackup CLI user role* topic of the *NetBackup Appliance Security Guide*.

Chapter 5

MSDP volume group (MVG)

This chapter includes the following topics:

- About the MSDP volume group
- Configuring the MSDP volume group
- MSDP volume group disaster recovery
- The MSDP server maintenance
- Limitations of the MSDP volume group
- About the node failure management
- MSDP volume group best practices
- MSDP commands for MVG maintenance
- Troubleshooting the MVG errors
- Examples of valid WORM MVG volumes
- About the client and policy assignments with disk volumes in MVG volume
- The modes of a disk volume in MVG volume
- Recommendation for managing the data distribution
- MVG load balancing levels
- Specifications and default values of the MVG tuning parameters
- CLI for tuning MVG parameters

About the MSDP volume group

A large NetBackup system usually has multiple MSDP servers. You may want to handle them together as one single pool to simplify the management.

The MSDP volume group (MVG) is an MSDP capability to build the volume groups in the storage layer on top of individual MSDP storage servers. One volume group is represented to NetBackup as a virtual volume, and this virtual volume is called an MVG volume. It can be used by NetBackup like a regular volume in a disk pool or storage unit configurations. The MSDP server that has the MVG capability and manages the MVG volumes is called an MVG server. It is a lightweight MSDP server.

The following are some of the advantages of the MSDP volume group:

- Groups the MSDP volumes to represent a large storage pool for ease of management.
- Improves system resilience with node failures.
- Manages the workload assignments and balancing.
  - Assignments for systems with a large number of clients at initial setup time
  - Assignments for newly added clients during the normal operation
  - Rebalance with storage server addition for capacity expansion
  - Reassignments to tolerate the storage failures
- Addresses the following storage unit group limitations:
  - Auto image replication (AIR)
  - Optimized duplication target support
- Supports almost all existing MSDP capabilities.
  See “Limitations of the MSDP volume group” on page 255.
- Works with the new or the existing MSDP servers.
- Lowers the total cost of ownership with single experience with NetBackup Appliance, Flex, and BYO.

See “Configuring the MSDP volume group” on page 236.

See “MSDP volume group disaster recovery” on page 252.

See “The MSDP server maintenance” on page 254.

See “Limitations of the MSDP volume group” on page 255.

See “About the node failure management” on page 256.

See “MSDP volume group best practices” on page 258.

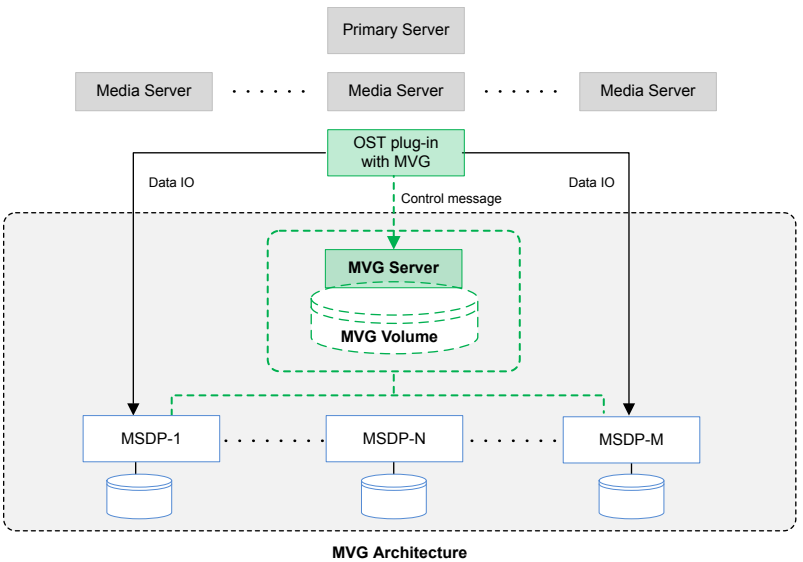
The MSDP volume group components

The MSDP volume group has the following two components:

- MVG server
  The MVG server is a lightweight MSDP server that has the MVG capability. One NetBackup domain can have multiple MVG servers.
  In NetBackup 10.5, it can run on any NetBackup media server on the Linux Red Hat platform. We recommend that you run it on a Flex multi-node appliance or along with the NetBackup primary server.
  See "About MVG resilience" on page 236.
- MVG volume
  An MVG volume is a virtual volume. It is an abstraction of a group of regular volumes from individual MSDP servers. One MVG server can manage multiple MVG volumes.
  It can be used by NetBackup in the same way as a regular volume in a disk pool and storage unit configurations.

The following diagram illustrates the architecture of the MSDP volume group.

Figure 5-1 MVG architecture



About MVG resilience

The MVG resilience is essential for maintaining system stability and preventing disruptions to backup processes. Following are the key aspects to consider for the MVG resilience:

- MVG server resilience
  The resilience of the MVG server is critical to overall system reliability. Here are key deployment strategies to ensure optimal resilience:
  - For BYO systems, deploy the MVG server with the NetBackup primary server. The resilience of both the MVG server and the NetBackup primary server is the same, regardless of whether the NetBackup primary server is clustered or not.
  - For Flex, deploy the MVG server on the same Flex host as the NetBackup primary server instance. Again, both servers have the same resilience level, regardless of whether it is Flex multi-node appliance or not.
- MVG node resilience
  Node-level resilience is managed by the system, such as Flex multi-node appliance, which provides fault tolerance at the node level.
- MVG data resilience
  Data resilience is also provided by the system, with technologies like RAID6 disk arrays providing redundancy and protection against data loss, ensuring the integrity of stored data.
- Backup job resilience
  In the event of a node failure without node-level resilience, the MVG server can redirect backup jobs to another node under certain conditions, minimizing the impact on backup operations.

These strategies ensure that the MVG server remains resilient, minimizing disruptions, and maintaining high availability for backup tasks.

Configuring the MSDP volume group

You can perform the following tasks to configure and use the MSDP volume group.

Table 5-1 Configure MSDP volume group

Task	Description
Ensure that the MSDP volume group requirements are met.	See "MSDP volume group requirements" on page 237.

Table 5-1 Configure MSDP volume group *(continued)*

Task	Description
Configure the MVG server.	See "Configuring an MVG server using the web UI" on page 239.
Configure the MVG volume.	See "Creating an MVG volume using the web UI" on page 239.
Configure an MVG server using the command-line	See "Configuring an MVG server using the command-line" on page 240.
Create an MVG volume using the command-line	See "Creating an MVG volume using the command-line" on page 242.
Update an MVG volume using the command-line	See "Updating an MVG volume using the command-line" on page 243.
Configure the targeted AIR with MVG volume.	See "Configuring the targeted AIR with an MVG volume " on page 244.
Update an MVG volume.	See "Updating an MVG volume using the web UI" on page 244.
List the MVG volumes.	See "Listing the MVG volumes" on page 245.
Delete an MVG volume.	See "Deleting an MVG volume" on page 245.
Configure the MSDP server to be used by MVG server if they have different credentials	See "Configuring the MSDP server to be used by an MVG server having different credentials" on page 248.
Migrate a backup policy to use the MSDP volume group.	See "Migrate a backup policy from a regular MSDP disk volume to the MVG volume" on page 250.
Migrate a backup policy from MVG to a regular MSDP disk volume.	See "Migrate a backup policy from an MVG volume to a regular MSDP disk volume" on page 250.
Assign a client policy combination to another MSDP server.	See "Assigning a client policy combination to another MSDP server" on page 251.
Remove the MVG server configuration.	See "Removing an MVG server configuration" on page 251.

MSDP volume group requirements

The requirements for an MVG server:

- An MVG server can run on any NetBackup media server on the Red Hat Enterprise Linux platform.
  - It can be a Flex media server container, a BYO media server, or a primary server.
  - It is recommended that you run an MVG server on a Flex multi-node appliance or along with the NetBackup primary server.
    See "About MVG resilience" on page 236.
  - It is not an MSDP container like that on Flex WORM, Access Appliance, NetBackup Flex Scale, or Cloud Scale.
  - An MVG server and MSDP server cannot exist on the same media server as an MVG server is a lightweight MSDP server.
- As the MVG server is new, it cannot be upgraded or migrated from a regular MSDP server.
- By default, one MVG server can have up to 64 MVG volumes, and each MVG volume can have up to 8 disk volumes.
- The disk volumes PureDiskVolume or cloud volumes cannot be configured on an MVG server.
- The MVG server requires hundreds of gigabytes of disk storage in a large environment.
- One NetBackup domain can have multiple MVG servers.

The requirements for the disk volumes of an MVG volume:

- The MSDP servers of the disk volumes must run with NetBackup 10.5 or later.
- The MVG server can use its default credentials to access the MSDP servers of the disk volumes.
- The disk volumes must be all PureDiskVolumes or all cloud volumes.
- The disk volumes must come from different MSDP servers.
- The disk volumes must have the same KMS, encryption, and WORM settings.
- If they are cloud volumes, they should be configured with comparative storage classes or tiers.
- The WORM MVG volumes must have overlapping ranges for retention locks. Within the WORM MVG volume, the shortest maximum lock duration for any WORM volume must be greater than the longest minimum lock duration for any WORM volume.
  See "Examples of valid WORM MVG volumes" on page 264.
- A disk volume cannot be assigned to multiple MVG volumes.

Configuring an MVG server using the web UI

You can configure an MVG server using the NetBackup web UI to use MSDP volume group.

To configure an MVG server

1. On the left, click **Storage > Disk storage**. Click the **Storage servers** tab, then click **Add**.
2. In the **Storage type** list, select **Disk storage servers**.
3. From the **Category** options, select **Media Server Deduplication Pool (MSDP, MSDP Cloud, MVG)**.

   Click **Start**.
4. In **Basic properties**, enter all the required information.

   To select your media server, click the search icon. If you do not see the media server you want to use, you can use **Search** field to find it.

   Click **Next**.
5. In the **Storage server** options, enter all required information and select **Enable MSDP volume group (MVG) service**.

   This option configures an MSDP server as an MSDP volume group (MVG) server, which lets you group the volumes from other MSDP servers to create MVG volumes. After you enable it, the MVG server can only host MVG volumes and cannot host its own local or cloud volumes.

   If you use Key Management Service (KMS), it must be configured before you can select the **KMS** option.

   Click **Next**.
6. (Optional) In **Media servers**, click **Add** to add any additional media servers that you want to use.

   Click **Next**.
7. On the **Review** page, confirm that all options are correct and click **Save**.

Creating an MVG volume using the web UI

You can configure an MVG volume using the NetBackup web UI.

To configure an MVG volume

1. On the left, click **Storage > Disk storage**.
2. Click **Storage > Disk storage**. Click the **Disk pools** tab, then click **Add**.

3. In the **Disk pool options**, click **Change** to select a storage server that is MVG enabled.
4. Enter all the required information. Click **Next**.
5. In **Volumes**, from the **Volume** drop down, select **Add MVG volume**.
6. Enter the MVG volume name and select the attributes to filter the desired volumes.
7. Select multiple volumes for the MVG volume. Click **Next**.
8. On the **Review** page, verify that all settings and information are correct. Click **Finish**.
9. After MVG volume is added, the MVG column appears under the **Disk pools** tab.

   Use the disk pool of the MVG volume in the same way as before to configure storage unit and replication targets.

Configuring an MVG server using the command-line

You can use the command-line to configure an MVG server on the Red Hat Linux media server.

See "MSDP commands for MVG maintenance" on page 259.

To configure the MSDP volume group server using the command-line

1. Create a storage server record in NetBackup.

   ```
   nbdevconfig -creatests -storage_server <media-server-fqdn> -stype
   PureDisk -media_server <media-server-fqdn> -st 9
   ```

2. Add MSDP credentials in NetBackup.

   ```
   /usr/openv/volmgr/bin/tpconfig -add -storage_server
   <media-server-fqdn> -stype PureDisk -sts_user_id <msdp-username>
   -password <msdp-password>
   ```

3. Get the MSDP configuration file template. The returned template should have a configuration item `mvgenabled 0`.

   ```
   nbdevconfig -getconfig -storage_server <media-server-fqdn> -stype
   PureDisk -configlist ./cfg.msdp.template
   ```

4 Create the MSDP configuration file in the following format.

Use the configuration file template `./cfg.msdp.template` to create the configuration file `./cfg.msdp`.

Change `mvgenabled: 0` to `mvgenabled: 1`.

The MSDP credentials in the configuration file must be the credentials that are added in NetBackup in the previous step.

For example,

```
# cat ./cfg.msdp
V7.5 "storagepath" "/sample-msdp-path" string
V7.5 "spalogin" "sample-username" string
V7.5 "spapasswd" "sample-password" string
V7.5 "spalogretention" "7" int
V7.5 "verboselevel" "3" int
V7.5 "dbpath" "/sample-msdp-path" string
V7.5 "required_interface" "" string
V7.5 "encryption" "1" string
V7.5 "mvgenabled" "1" string
```

It is recommended to specify the same path for MSDP storage and catalog with `storagepath` and `dbpath`.

5 Initialize the MSDP server.

```
nbdevconfig -setconfig -storage_server <media-server-fqdn> -stype
PureDisk -configlist ./cfg.msdp
```

On success, `mvg-controller` and `mvg-mds` along with the other MSDP services are initialized and started.

Creating an MVG volume using the command-line

To configure an MVG volume using the command-line

1 Create a configuration file for an MVG volume in the following format.

```
V7.5 "operation" "add-virtual-volume" string
V7.5 "virtualVolume" "<mvg-volume>" string
V7.5 "diskVolume" "<msdp-server1>:<msdp-volume1>:Y" string
V7.5 "diskVolume" "<msdp-server2>:<msdp-volume2>:Y" string
V7.5 "diskVolume" "<msdp-server...>:<msdp-volume...>:Y" string
```

For example,

```
# cat sample-mvg-local.cfg
V7.5 "operation" "add-virtual-volume" string
V7.5 "virtualVolume" "sample-mvg-local" string
V7.5 "diskVolume" "sample-msdp-server1:PureDiskVolume:Y" string
V7.5 "diskVolume" "sample-msdp-server2:PureDiskVolume:Y" string
V7.5 "diskVolume" "sample-msdp-server3:PureDiskVolume:Y" string
V7.5 "diskVolume" "sample-msdp-server4:PureDiskVolume:Y" string

# cat sample-mvg-cloud.cfg
V7.5 "operation" "add-virtual-volume" string
V7.5 "virtualVolume" "sample-mvg-cloud" string
V7.5 "diskVolume" "sample-msdp-server1:cloud-volume1:Y" string
V7.5 "diskVolume" "sample-msdp-server2:cloud-volume2:Y" string
V7.5 "diskVolume" "sample-msdp-server3:cloud-volume3:Y" string
V7.5 "diskVolume" "sample-msdp-server4:cloud-volume4:Y" string
```

2 Create an MVG volume in the MVG server.

```
nbdevconfig -setconfig -storage_server <mvg-server> -stype
PureDisk -configlist <configuration-file>
```

3 Create a disk pool with the MVG volume in NetBackup.

```
nbdevconfig -previewdv -storage_server <mvg-server> -stype
PureDisk | grep <mvg-volume> > <mvg-volume-cfg-file>
```

```
nbdevconfig -createdp -dp <disk-pool-name> -storage_server
<mvg-server> -stype PureDisk -dvlist <mvg-volume-cfg-file>
```

Updating an MVG volume using the command-line

To update an existing MVG volume using the command-line

1 Create a configuration file to update an MVG volume in the following format.

```
V7.5 "operation" "update-virtual-volume" string
V7.5 "virtualVolume" "<mvg-volume>" string
V7.5 "diskVolume" "<msdp-server1>:<msdp-volume1>:Y" string
V7.5 "diskVolume" "<msdp-server2>:<msdp-volume2>:Y" string
V7.5 "diskVolume" "<msdp-server...>:<msdp-volume...>:Y" string
V7.5 "diskVolume" "<new-msdp-server1>:<new-msdp-volume1>:Y" string
V7.5 "diskVolume" "<new-msdp-server...>:<new-msdp-volume...>:Y"
string
```

For example,

```
# cat sample-mvg-local.cfg
V7.5 "operation" "update-virtual-volume" string
V7.5 "virtualVolume" "sample-mvg-local" string
V7.5 "diskVolume" "sample-msdp-server1:PureDiskVolume:Y" string
V7.5 "diskVolume" "sample-msdp-server2:PureDiskVolume:Y" string
V7.5 "diskVolume" "sample-msdp-server3:PureDiskVolume:Y" string
V7.5 "diskVolume" "sample-msdp-server4:PureDiskVolume:Y" string
V7.5 "diskVolume" "sample-msdp-server-new1:PureDiskVolume:Y" string
V7.5 "diskVolume" "sample-msdp-server-new2:PureDiskVolume:Y" string
```

Example to specify an alternative storage server for one or more MVG volumes.

```
# cat sample-mvg-local1.cfg
V7.5 "operation" "update-virtual-volume" string
V7.5 "virtualVolume" "sample-mvg-local" string
V7.5 "diskVolume"
"sample-msdp-server1:PureDiskVolume:Y:altServer=<alt STS1>" string
V7.5 "diskVolume"
"sample-msdp-server2:PureDiskVolume:Y:altServer=<alt STS2>" string
V7.5 "diskVolume" "sample-msdp-server3:PureDiskVolume:Y" string
V7.5 "diskVolume" "sample-msdp-server4:PureDiskVolume:Y" string
V7.5 "diskVolume" "sample-msdp-server-new1:PureDiskVolume:Y"
string
V7.5 "diskVolume" "sample-msdp-server-new2:PureDiskVolume:Y"
string
```

2 Apply the configuration changes on the virtual volume.

```
nbdevconfig -setconfig -storage_server <mvg-server> -stype
PureDisk -configlist <configuration-file>
```

Configuring the targeted AIR with an MVG volume

MSDP volume group supports targeted AIR only. The legacy untargeted AIR is not supported.

You must set up secure communication on the MSDP servers that are used by an MVG volume. These MSDP servers connect to the target MVG volumes or MSDP servers for the data replication.

Before configuring the replication targets on NetBackup web UI, run the following commands on those MSDP servers and the MVG server.

```
nbcertcmd -getCACertificate -server <target-primary-server>
```

```
nbcertcmd -getCertificate -server <target-primary-server> -token
<auth-token>
```

Configuring replication targets with MSDP volume group is the same as regular MSDP.

See "About NetBackup Auto Image Replication" on page 139.

Updating an MVG volume using the web UI

An MVG supports append-only updates, meaning you cannot insert, delete, or reorder the disk volumes in the group. However, you can perform the following actions through the web UI:

- Set or update the alternative storage server by editing and saving the MVG configuration.
- Change the mode of a disk volume within the MVG, such as switching it to the read-only mode.

Note: If a disk volume fails and cannot be recovered, you can change its mode to read only in the MVG volume. Alternatively, you can re-create the MVG volume by removing it from the list of the disk volumes.

The alternative storage server is used only when the original node is down. In such cases:

- The cloud LSU on the alternate node must be manually mounted in read-only mode.

- Once mounted, the alternate node can access pre-existing data and new data that is written by the original node after the LSU was mounted.

For instructions on mounting a cloud LSU as read only on a storage server, see the following topic.

See "Reuse cloud LSU as read-only" on page 335.

The alternative storage server supports restore, verification, and optimized duplication operations.

Note: The alternative storage server must be a valid MSDP storage server within the current domain.

Listing the MVG volumes

To list the MVG volumes

- Run the following command on the MVG server to dump the MVG volume configuration information.

```
cacontrol --mvg listvvols [<dsid-of-mvg-volume>]
```

Deleting an MVG volume

An MVG volume is not deleted from the MVG server when the corresponding disk pool on NetBackup web UI is deleted.

To delete an MVG volume

1. (Optional) Take a backup of the MSDP volume group configuration. Run the following command on the MVG server to dump the MVG volume configuration information to a file.

   ```
   cacontrol --mvg listvvols [<dsid-of-mvg-volume>]
   ```

2. Create the configuration file for the MVG volume deletion in the following format.

   ```
   V7.5 "operation" "delete-virtual-volume" string
   V7.5 "virtualVolume" "<mvg-vol>" string
   ```

3. Delete an MVG volume in the MVG server.

   ```
   nbdevconfig -setconfig -storage_server <mvg-server> -stype
   PureDisk -configlist <configuration-file>
   ```

   This command removes an MVG volume association information from the MSDP servers of the disk volumes and removes an MVG volume information from the MVG server. Ensure that the MSDP servers are accessible when you run this command.

   If an MSDP server is down for some reason, the command does not fail. Do not use this command if an MSDP server is down as the MVG volume associated information is not removed from that MSDP server.

   Run the following commands on that MSDP server to manually remove the information from it when it is back.

   ```
   cacontrol --dataselection removefromvvol <dsid-of-disk-volume>
   <disk-volume> <msdp-server> <mvg-server> <mvg-volume>
   ```

   ```
   cacontrol --dataselection refresh-disk-volume <msdp-server>
   <disk-volume >
   ```

Deleting a disk volume from an MVG volume configuration

Before removing a disk volume from an MVG volume, make sure that the data is safely evacuated. You can use the MVG evacuation workflow to evacuate the data.

See “About MSDP server evacuation” on page 273.

Once evacuation is complete, you can o delete the disk volume from the MVG volume configuration using NetBackup web UI or command line.

If the disk volume is already lost or unrecoverable, you can forcibly remove it.

Ensure that all other disk volumes in the MVG volume are healthy and accessible before starting.

To delete a disk volume from an MVG volume

1. Create a configuration file to shrink the MVG volume in the following format.

   Create a configuration file in the following format. Specify only one disk volume to remove. If it's the only disk volume, follow the instructions to delete the entire MVG volume instead. See "Deleting an MVG volume" on page 245.

   Sample format:

   ```
   V7.5 "operation" "shrink-virtual-volume" string
   V7.5 "virtualVolume" "<mvg-volume-name>" string
   V7.5 "diskVolume" "<msdp-server>:<msdp-volume>:Y" string
   ```

   Example:

   ```
   cat /root/delete-host4-from-mvg-local.cfg
   V7.5 "operation" "shrink-virtual-volume" string
   V7.5 "virtualVolume" "mvg-local" string
   V7.5 "diskVolume" "mvgdemo-host4.mvgdemo.com:PureDiskVolume:Y" string
   ```

2. Apply the configuration.

   ```
   nbdevconfig -setconfig -storage_server <mvg-server> -stype
   PureDisk -configlist <configuration-file>
   ```

3. Confirm Disk Volume removal. Run the following command or verify through the **MVG volume details** page in the web UI.

   ```
   cacontrol --mvg listvvols
   ```

Identifying the orphan images stored on the lost disk volume

If a disk volume is lost and cannot be recovered, resulting in data loss, you can forcibly remove it from the MVG volume. This process also includes identifying and cleaning up the orphaned backup images that remain in the NetBackup catalog but are no longer accessible.

Ensure that all other disk volumes in the MVG configuration are functional and accessible.

To identify the orphan images stored on the lost disk volume

1. Run the following command on the MVG server or a load balancing media server to list the images that are associated with the MVG volume:

   ```
   /usr/openv/netbackup/bin/admincmd/bpstsinfo -ii -stype PureDisk
   -storage_server <mvg-server> -lsuname <mvg-volume>
   ```

2. List the images from the NetBackup catalog.

   Run the following command on the MVG server, a load-balanced media server, or the primary server. Use a sufficiently old start date to ensure that all backup images are included:

   ```
   /usr/openv/netbackup/bin/admincmd/bpimmedia -d 01/01/2000 -stype
   PureDisk -dp <mvg-dp> -dv <mvg-volume>
   ```

3. Identify the orphaned images.

   Compare the results from the previous steps to identify the orphaned images that are present in the NetBackup catalog but missing from the MVG volume, and carefully verify the accuracy of the list.

4. Run the following command to delete the orphan images from the NetBackup catalog.

   ```
   /usr/openv/netbackup/bin/admincmd/bpexpdate -backupid <backup_id>
   -copy <copy_number> -nodelete -force -d 0 -force_not_complete
   -do_not_follow_dependee [-try_expire_worm_copy]
   ```

   You can contact Cohesity Support for assistance with cleaning up orphaned images.

Configuring the MSDP server to be used by an MVG server having different credentials

To add a disk volume of an MSDP server to an MVG volume, the MVG server must be able to use its default credentials to communicate with the MSDP server.

If the MSDP server is new, configure the MSDP server with the same credentials as the MVG server so that no additional configuration change is required on the MSDP server.

In many cases, especially if the MSDP server already exists, the credentials may be different. You must change the existing credentials or add new credentials on the MSDP server.

To configure the MSDP server to be used by MVG server if the passwords are different

- Follow the MSDP existing instructions to change the password.

  How to reset the NetBackup authentication password on an MSDP disk pool.

To configure the MSDP server to be used by MVG server if the usernames are different

1. Add an alias for the MSDP default user. The username and password of the alias user are the same as the MVG server credentials.

   Windows: `<install_path>\pdde\spauser --a -u username [-p password] --owner admin –alias`

   UNIX: `/usr/openv/pdde/pdcr/bin/spauser -a -u username [-p password] --owner admin –alias`

   Note: Adding an alias user is supported on NetBackup 10.5 or later. Do not run the command on an MSDP server with an earlier version.

2. After you add an alias user, ensure that the alias user has the same data selection ID as the admin user. Run the following command:

   ```
   spauser -l
   ```

   For example,

   ```
   root@host ~ # spauser -l
   user 1 : msdp, data selection: 2, role: admin
   root@host ~ # spauser -a -u mvg-user --owner msdp --alias

   Password:
   Reenter Password:
   Please input password of super user [msdp]:

   Add user mvg-user successful !

   One MSDP user can only be used by one NBU domain! Do not use one
    same user in two or more NBU domains!
   root@host ~ # spauser -l
   user 1 : msdp, data selection: 2, role: admin
   user 2 : mvg-user, data selection: 2, role: admin
   ```

See “Adding MSDP admin alias users from the deduplication shell” on page 712.

Migrate a backup policy from a regular MSDP disk volume to the MVG volume

Before the migration, we recommend assigning the MSDP disk volume that policy uses to the MVG volume that the policy will use. This step is especially important for an accelerator backup policy. The MVG server continues using the previous disk volume and the previous Accelerator track log for the new backup jobs. The deduplication ratio and backup performance are affected minimally.

If the backup policy used the multiple MSDP disk volumes and the backup images still exist on those MSDP servers, assign at least the disk volume that is most recently used to the MVG volume. You can also assign all the MSDP disk volumes to the MVG volume.

Run the following MSDP command to specify the most recently used one as the primary one.

```
cacontrol --cluster set-cp-assignment <dsid-of-mvg-volume> <client>
<policy> <msdp-server>
```

Migrate the backup policies gradually overtime to reduce any potential impact.

Migrate a backup policy from an MVG volume to a regular MSDP disk volume

It is recommended to change to the MSDP disk volume that is most recently used by an MVG volume for the policy. You can find the information on the Notification webpage on NetBackup web UI. To find the information by using the MSDP command-line, run the following command:

```
cacontrol --cluster get-cp-assignment <dsid-of-mvg-volume> <client>
<policy>
```

You can also find the information in the job details of the recent backup jobs.

Migrate the backup policies gradually overtime to reduce any potential impact.

Assigning a client policy combination to another MSDP server

To assign a client policy combination to another MSDP server

1. Run the following MSDP command to check which MSDP servers the client policy combinations are currently assigned to.

   ```
   cacontrol --cluster get-cp-assignment <dsid-of-mvg-volume>
   <client> <policy>
   ```

2. Run the following MSDP command to change the assignments:

   ```
   cacontrol --cluster set-cp-assignment <dsid-of-mvg-volume>
   <client> <policy> <msdp-server>
   ```

Removing an MVG server configuration

Perform the following steps to remove an MVG server configuration.

To remove an MVG server configuration

1. On NetBackup, remove all the configuration of the corresponding storage unit, disk pool, and storage server with an MVG server.
2. Delete the MVG volumes from the MVG server.

   Note: Do not delete the MVG volumes if you want to preserve the MVG data or want to reconfigure the MVG server by reusing the existing data.

3. Stop the NetBackup services on the host.

   ```
   bp.kill_all
   ```

4. Delete the storage server data on the disk.

   - Run the following script.

     ```
     /usr/openv/pdde/pdconfigure/scripts/installers/PDDE_deleteConfig.sh
     ```

     Type **y** and press **Enter**.
     The configuration file is restored to the original state.
   - Delete the data in the MSDP storage path and database path.

     Note: Do not perform this step if you want to preserve the MVG data or want to reconfigure the MVG server by reusing the existing data.

5. Restart the NetBackup services on the host.

   ```
   bp.start_all
   ```

MSDP volume group disaster recovery

You can protect the MSDP catalog of an MVG server as a regular MSDP server. You can use an MSDP catalog backup policy.

Scenario 1: MSDP catalog can be recovered

If the MSDP catalog can be recovered, you can follow the same steps to reconfigure the MSDP server with the recovered catalog.

We recommend that you protect the MSDP catalog of an MVG server as a regular MSDP server. One way is to use an MSDP catalog backup policy.

Scenario 2: MSDP catalog cannot be recovered from MVG server catalog backup

For some reason, the MSDP catalog cannot be recovered because there is no backup or the backup data cannot be used. You can rebuild the MSDP volume group configuration data and reconfigure the MVG server.

Table 5-2

No	Step	Description
1	Reassemble the MSDP volume group configuration.	Find out the MVG volumes that MVG server had and the physical volumes each MVG volume used. Use one of the following ways to find out the information. ■ Use the `GET /storage/storage-servers/{storageServerId}/disk-volumes` API If a physical volume is used by an MVG volume, the physical volume has an attribute `usedByVLsu=<mvg-server>:<mvg-volume>` in the returned data. If it is an MVG volume, `diskVolumeCapabilities` list has the keyword `PUREDISK_VIRTUAL` in the attributes in the returned data. ■ Run the following NetBackup command: `nbdevquery -liststs -stype PureDisk` `bpstsinfo -li` ■ Run the following MSDP command on each MSDP server: `/usr/openv/pdde/pdcr/bin/cacontrol --dataselection list`

Table 5-2 *(continued)*

No	Step	Description
2	Reconfigure an MVG server.	You can configure the MSDP server with MVG option selected like an MSDP server. Using the previous host name is recommended but is not necessary. If NetBackup lost the configuration of the storage server for the MVG server, use NetBackup web UI to configure it. If NetBackup still has the configuration of the storage server for the MVG server, use `nbdevconfig -setconfig` command. The configuration list file has no difference with that for a regular MSDP server, except we need to enable MVG by adding: `V7.5 "mvgenabled" "1" string` For more information, See "Configuring an MVG server using the command-line" on page 240.
3	Add back the MVG volumes one by one.	If NetBackup still has the disk pool configured with the MVG volumes, run NetBackup command `nbdevconfig -setconfig` with `add-virtual-volume` option to add back the MVG volumes one by one, with the information collected in Step 1. See "Creating an MVG volume using the command-line" on page 242. In case the disk pool configuration with the MVG volumes was lost, use NetBackup web UI to recreate the corresponding disk pools and MVG volumes with the information collected in Step 1. In this case, NetBackup catalog also most likely lost the backup image records, which are associated with the MVG volumes. Thus, like recovering a regular MSDP server for the physical volumes, NetBackup image 2-phase import is also needed after the MVG server and MVG volumes are reconfigured.

Table 5-2 *(continued)*

No	Step	Description
4	(Optional) Adjust the client and policy assignment table.	The client and policy assignment table is rebuilt in the step 3. If a client and policy combination were taken by multiple MSDP servers before disaster, the information can be rebuilt but the primary one may be different than the one before the disaster. Run the following MSDP commands to check and adjust the primary MSDP server of a client and policy combination if required: `cacontrol --cluster get-cp-assignment <dsid-of-mvg-volume> [<client> [<policy>]]` `cacontrol --cluster set-cp-assignment <dsid-of-mvg-volume> <client> <policy> <msdp-server>` The MSDP server which has more data for the client and policy is usually set as the primary one.

The MSDP server maintenance

In a routine node maintenance such as upgrade, EEB installation, OS patching, and system restart we recommend that you intervene to reduce the impact on deduplication ratio or a job failure.

Use MSDP command-line of the MVG server or NetBackup web UI to set the node in the maintenance mode before the routine maintenance and reset afterwards. The MVG server does not move the client policy assignment from the node in maintenance to another node. The new backup jobs may fail temporarily if the node is not stable or functional during the maintenance.

Run the following MSDP commands on the MVG server:

```
cacontrol --mvg set-mvg-maintenance <msdp-server>
```

```
cacontrol --mvg unset-mvg-maintenance <msdp-server>
```

```
cacontrol --mvg get-mvg-maintenance
```

Example:

```
PATH=$PATH:/usr/openv/pdde/pdcr/bin
cacontrol --mvg set-maintenance-mode vramvg037340.rsv.ven.veritas.com
Updated virtual volumes: mvg-local
cacontrol --mvg get-maintenance-mode
```

```
MSDP servers in maintenance mode: vramvg037340.rsv.ven.veritas.com
Virtual volumes which have the MSDP servers in maintenance mode:
mvg-local
cacontrol --mvg unset-maintenance-mode
vramvg037340.rsv.ven.veritas.com
Updated virtual volumes: mvg-local
cacontrol --mvg get-maintenance-mode
No MSDP servers found in maintenance mode.
```

On the NetBackup web UI, go to the Disk Pool webpage of an MVG volume, select and edit a disk volume of an MVG volume, then switch the mode between "Maintenance" and "Normal".

Alternatively, you can deactivate the corresponding backup policies temporally during the node maintenance.

Alternatively, if the maintenance cycle is short, you can use or tune out the MVG rebalance freezing time on the MVG server, so that the MVG server does not move the client policy assignment out if the node can be back to stable within the MVG rebalance freezing time. By default, the value is 0.5 hour. It can be tuned by MVG Controller API:

```
curl -K /uss/config/certificates/controller/.curl_config
https://localhost:10100/v1/agent/mvg/loadbalance/config -d
'{"asmt_rb_freezing_timeout": 3600 }' -X PATCH
```

Check the tuning parameters:

```
curl -K /uss/config/certificates/controller/.curl_config
https://localhost:10100/v1/agent/mvg/loadbalance/config
```

Limitations of the MSDP volume group

The MSDP volume group does not support the following:

- Supports an isolated recovery environment (IRE) with the following configuration:
  - The clock time and time zone on the MSDP servers must be consistent with the MVG server, and both should run on Linux platforms.
  - One MSDP server should not be used by multiple MVG servers.
- Does not support universal share. You can use the individual MSDP servers for universal share as it is.
- Supports Image sharing on regular MSDP servers. For the backup images, which were created through MVG volume, you can use regular MSDP server as it is for image sharing.

- No deduplication across nodes.
- Limited data balancing across nodes.
  - To reduce the impact on deduplication rate, we restrict the reassignment of one client policy on a limited number of nodes. Manually changing the assignment of one client policy is supported. You can use an MSDP command to change that on a limited number of nodes.
  - A single large backup cannot spread across nodes.
    If it is an MVG volume of local volumes (PureDiskVolumes), it may be seen as an issue as there is space for the MVG volume, but not enough space for any single node to store the backup data.
    When the issue happens, as an alternative, you can move the large backup policy to an MVG volume of cloud volumes, or use "MSDPLB+" policy and enable multistream backup.
  - The backup jobs of a multistream backup policy will go to the same node unless it is an "MSDPLB+" policy. Do not use MSDPLB+ policy unless necessary as there is no data deduplication across the MSDP nodes.

    Note: "MSDPLB+" policy means a backup policy of which the name starts with "MSDPLB+". Starting from NetBackup 11.0, it also means a backup policy with the policy attribute **Enable Multiple MSDP nodes** is enabled.

About the node failure management

The following table lists the node failure scenarios and how they can be managed.

Table 5-3

Node failure	Description
Planned node failure	In a routine node maintenance, we recommend that you reduce the impact on deduplication ratio or a job failure. Set the node in a maintenance mode on an MVG server. The client policy assignments on the node are not automatically changed if the node is in maintenance mode. Run the following MSDP command to change them manually: `cacontrol --mvg set-maintenance-mode <msdp-server>` `cacontrol --mvg unset-maintenance-mode <msdp-server>` `cacontrol --mvg get-maintenance-mode` On the NetBackup web UI, go to the Disk Pool webpage of an MVG volume, select and edit a disk volume of an MVG volume, then switch the mode between "Maintenance" and "Normal".
Unplanned node failure	The MVG server uses a rebalance freezing timeout to control when it reassigns disk volumes. If the default disk volume becomes unavailable, the MVG server will not select another volume for the same client-policy until the default volume remains unavailable beyond the configured timeout during the next backup. This setting is configurable using the tuning parameter `asmt_rb_freezing_timeout` through the MVG tuning API or the command line. See “Specifications and default values of the MVG tuning parameters” on page 270. See “CLI for tuning MVG parameters” on page 272.
The node is operational again after the failure	If a node stays down for long, the client policy assignments are moved out when NetBackup run backup jobs with the backup policies. When the node is back, most of the original client policy combinations are moved back to keep the system balanced again.

Table 5-3 *(continued)*

Node failure	Description
The node is full	If a disk volume is full, a client policy is moved to another node. It is not moved back unless the new node becomes inactive or full and the original node has space available again. The setting can be disabled using the tuning parameter `asmt_rb_freezing_on_disk_full` through the MVG tuning API or the command line. See "Specifications and default values of the MVG tuning parameters" on page 270. See "CLI for tuning MVG parameters" on page 272.

MSDP volume group best practices

The following are some of the best practices with the MSDP volume group.

- Put an MVG server close to the regular MSDP servers of which the physical volumes are grouped by the MVG server.
  The MVG server and the load balance media servers of the MVG server can reach the MSDP servers efficiently with low latency and good throughput.
- Add the media servers of the regular MSDP servers to the load balance media server list of the MVG server. Consider adding more media servers to the list if there are.
- Put an MVG server in a reliable place, which is desired to have physical isolation with the regular MSDP servers.
  As an example, the MVG server can be configured on the NetBackup primary server if it is the only MVG server, or on HA-enabled Flex appliance or BYO environment.
- The physical volumes in one MVG volume have the similar configuration.
  They have similar volume size, similar server CPU/memory, similar disk/network performance. It makes a more balanced load between the physical volumes and MSDP servers.
  They are enforced to have the same encryption, KMS, WORM, and other security configuration. The MVG volume creation fails if the settings do not match.
- Add a new MVG server only when needed to reduce cost and not compromise on the performance.
  Add a new MVG server only when the existing ones cannot meet the requirements. For example, physical isolation between the MVG servers and

the MVG volumes is needed. The existing MVG server too small to manage many MVG volumes. The MVG server has too many MVG volumes.

- Do not use multiple MVG servers to manage the physical volumes of the same MSDP servers.
  Do not have a physical volume assigned to multiple MVG volumes, does not matter if the MVG volumes are managed by the same or different MVG servers.
  If an MSDP server has multiple volumes, they can be assigned to different MVG volumes of different MVG servers. However, it is not recommended.
- Once a physical volume is assigned to an MVG volume, it's not recommended to use it anymore for NetBackup jobs.
  It does not block anything if the physical volume continues being used. You can still use it to restore the pre-existing backup images. The currently working configuration with the physical volume can continue working. To have a more balanced load, remove it from the configuration of the backup policy when it is assigned to an MVG volume.
- Deactivate the media server on an MVG server if it is small.
  If the MVG server is small, deactivate the media server on the MVG server to prevent NetBackup from scheduling jobs on it when there are additional load-balancing media servers for the MVG server. Do not deactivate it if it is the only load-balancing media server of the MVG server.
  You can go to NetBackup web UI **Storage > Media server**, select the MVG server, then click **Deactivate**.
  You can also run the following NetBackup command:
  ```
  /usr/openv/netbackup/bin/admincmd/nbemmcmd -updatehost -machinename
  <mvg-server> -machinestateop set_admin_pause -machinetype media
  -masterserver <primary-server>
  ```
- Create the media server and MSDP engine affinity.
  See "Enabling the media server and MSDP engine affinity in the MSDP cluster" on page 570.

MSDP commands for MVG maintenance

The following table lists the MSDP commands that you can use to manage the MVG configuration.

Table 5-4 MSDP commands for an MVG

Task	Command
List the data selections of MSDP server.	Run the following command on the MSDP or the MVG server. `cacontrol --dataselection list`
Get the MSDP disk volume configuration.	Run the following command on the MSDP or the MVG server. `cacontrol --dataselection getlsuconfig`
Refresh the MSDP disk volume of an existing NetBackup disk pool.	Run the following command on the MSDP server. `cacontrol --dataselection refresh-disk-volume <msdp_server> <volume-name>`
Add the MVG volume association state on an MSDP disk volume.	Run the following command on the MSDP server. `cacontrol --dataselection assigntovvol <dsid-of-disk-volume> <disk-volume> <msdp-server> <mvg-server> <mvg-vol>` The MVG controller invokes the command for each MSDP server and each physical volume when an MVG volume is created or updated. Use this command only for debugging and troubleshooting purposes.
Remove the MVG volume association state on an MSDP disk volume.	Run the following command on the MSDP server. `cacontrol --dataselection removefromvvol <dsid-of-disk-volume> <disk-volume> <msdp-server> <mvg-server> <mvg-vol>` The MVG controller invokes the command for each MSDP server and each physical volume when an MVG volume is deleted or the creation or update of an MVG volume is cancelled. Use this command only for debugging and troubleshooting purposes.

Table 5-4 MSDP commands for an MVG *(continued)*

Task	Command
Get which MSDP server the client and the policy combinations are assigned to.	Run the following command on the MVG server. `cacontrol --cluster get-cp-assignment <dsid-of-mvg-volume> [<client> [<policy>]]`
Assign client and policy combination to an MSDP server.	Run the following command on the MVG server. `cacontrol --cluster set-cp-assignment <dsid-of-mvg-volume> <client> <policy> <msdp-server>`
Find the MSDP catalog.	Run the following command on the MSDP server. `cacontrol --catalog find <dsid-of-disk-volume> <dirname> <basename> [ --listtype ALL|DV_ONLY|VV_ONLY ]` `VV_ONLY`: Filter the path objects, which were created through MVG. `DV_ONLY`: Filter the path objects, which were created not through MVG. For example: `/usr/openv/pdde/pdcr/bin/cacontrol --catalog find 2 / "*" --listtype VV_ONLY`
List the MVG volumes on MVG server.	Run the following command on the MVG server. `cacontrol --mvg listvvols`
Set the MSDP server in maintenance mode.	Run the following command on the MVG server. `cacontrol --mvg get-maintenance-mode | set-maintenance-mode <msdp_server> | unset-maintenance-mode <msdp_server>`

Table 5-4 MSDP commands for an MVG *(continued)*

Task	Command
Remotely find the MSDP catalog of the regular MSDP servers on MVG server.	Run the following command on the MVG server. `cacontrol --mvg catalog-find <dsid-of-mvg-volume> <dirname> <basename> [ --listtype VV_ONLY|DV_ONLY|ALL ] [ --hostname <msdp_server> ]`
Validate the MSDP server credentials.	Run the following command on the MSDP or the MVG server. `/usr/openv/pdde/pdcr/bin/spauser -v -stdin` When you are prompted, type `-u <username> -p <password> --role admin`.

Troubleshooting the MVG errors

The following table lists the MVG error messages and how to troubleshoot those.

Table 5-5 The MVG error messages

Error message	Description
Failed to validate the name: <detailed info>.	A specified volume name does not meet the MSDP requirements. See "NetBackup naming conventions" on page 26.

Table 5-5 The MVG error messages *(continued)*

Error message	Description
Listing the data selections failed. OR **Checking virtual volume awareness failed.**	The error occurs if the MVG server cannot use its default credentials to communicate with some MSDP server of the MSDP disk volumes of the MVG volume. You must change the credentials of the MSDP server. See "Configuring the MSDP server to be used by an MVG server having different credentials" on page 248. The error can also occur if the MVG server cannot connect to the MSDP server. Ensure that the connection is established and the MSDP server is functional. The error can also occur if the MSDP server is running with the older version of NetBackup. Ensure that the MSDP server is running with NetBackup 10.5 or later.
Conflict in the configuration on <description>. **Note:** <description> can be "encryption", "KMS", "WORM", "need Warming (for cold cloud storage)" and so on.	The disk volumes for the MVG volume do not have the compatible settings on WORM, Encryption, KMS, or storage class types if they are cloud volumes.
Volume <msdp-server>:<volume-name> (dsid:<dsid>) is inactive.	The MSDP server of one disk volume is unreachable or it is an inactive cloud volume.
Members <msdp-server>:<volume-name1> and <msdp-server>:<volume-name2> have the same server name.	Multiple disk volumes of the same MSDP server are selected to be assigned to one MVG volume.
Virtual volume already exists.	A duplicate MVG volume name is specified.
Including both cloud LSU and local LSU in a virtual volume is not expected.	Add a cloud volume to an MVG volume with PureDiskVolumes or add a PureDiskVolume to an MVG volume with cloud volumes.
Member number (<#>) of the MVG volume exceeds the maximum allowed number (<#>).	This error occurs when you try to assign too many disk volumes to an MVG volume. By default an MVG volume can have up to 8 disk volumes. The number is configurable but is not recommended.

Table 5-5 The MVG error messages *(continued)*

Error message	Description
Maximum allowed MVG volume number (<#>) is reached.	This error occurs when you try to create too many MVG volumes on the MVG server. By default an MVG server can have up to 64 MVG volumes. The number is configurable but is not recommended.
It's not allowed to reduce the member number from <#> to <#> for virtual volume <volume-name>.	This error occurs when you try to remove some disk volumes from an MVG volume.
It's not allowed to change disk volume <msdp-server1>:<volume-name1> to <msdp-server2>:<volume-name2> for virtual volume %s.	This error occurs when you try to insert a disk volume or change the order of the disk volume list of an MVG volume.
Member <msdp-server>:<volume-name> is not aware of virtual volume.	This error occurs when you try to assign a disk volume of an old MSDP server to an MVG volume.
Cannot find dsid for member <msdp-server>:<volume-name>.	A bad disk volume name is specified.
Changing virtual volume assignment failed: No disk pool was found for the disk volume.	The disk volume that is being assigned to an MVG volume does not have a corresponding disk pool configured in NetBackup.
Changing virtual volume assignment failed: failed to find the disk pool name <dp-name>.	NetBackup web service is not working normally. When the error happens, retry the MVG volume configuration or ensure that the NetBackup web service is working correctly.

Examples of valid WORM MVG volumes

A WORM MVG is made up of multiple individual WORM volumes. All individual volumes in the group must share the same WORM lockdown mode. They must all be in the "Compliance" mode or "Enterprise/Governance" mode.

Additionally, all the individual volumes must have overlapping minimum and maximum lock durations. Within a WORM MVG volume, the shortest maximum lock duration for any WORM volume must be greater than the longest minimum lock duration for any WORM volume.

The following table lists the valid and invalid WORM MVG volumes.

Table 5-6 WORM MVG volumes

Volume	WORM mode	Minimum Lock duration	Maximum Lock duration
WORM_VOL_1	Compliance	10 days	25 days
WORM_VOL_2	Compliance	5 days	30 days
WORM_VOL_3	Compliance	15 days	40 days
WORM_VOL_4	Enterprise	10 days	30 days
WORM_VOL_5	Compliance	30 days	60 days

In this table, WORM volumes 1, 2, and 3 can be members of the same MVG volume because they all share the same WORM mode (Compliance). The longest minimum lock duration is 15 days while the shortest maximum lock duration is 25 days. This creates an overlapping lock duration of 15-25 days, making this valid combination.

WORM volumes 1, 2 and 4 cannot be members of the same MVG volume because all of them do not have the same WORM mode. WORM_VOL_4 is in the Enterprise mode while the others are in the Compliance mode.

WORM volumes 1, 2 and 5 cannot be members of the same MVG volume because they do not have overlapping lock duration. The longest minimum lock duration is 30 days while the shortest maximum lock duration is 25 days. Since 30 days minimum lock duration is greater than 25 days maximum lock duration, there is no overlap, making this combination invalid.

About the client and policy assignments with disk volumes in MVG volume

MVG server uses client and policy combinations to distribute backup data across disk volumes. One combination can be assigned to one or multiple disk volumes with a priority order. The default disk volume is called the primary node, and others are called historic nodes.

When a client and policy combination is used for the first time, MVG selects a disk volume based on certain criteria. See See "MVG load balancing levels" on page 269. The subsequent jobs continue with the same client and policy combination using the selected disk volume unless it becomes unavailable or full.

If the primary node becomes unavailable, MVG does not immediately switch to another volume during the next backup job. Instead, it waits for a predefined duration

to determine whether the outage is temporary. This delay may result in job failures before the threshold is reached, but it helps avoid unnecessary data duplication and preserves deduplication efficiency. Once MVG decides to use a different volume, it selects the next available historic node if one exists. If not, a new disk volume is assigned as a historic node for that client/policy combination. When the original primary node becomes available again, MVG resumes using it for future jobs.

See "About the node failure management" on page 256.

If the primary node becomes full, MVG immediately switches to the next available historic node or assigns a new disk volume as the new primary node. The previously full volume becomes a historic node. This is the only case where the primary node is changed automatically.

See "Specifications and default values of the MVG tuning parameters" on page 270.

For VLDB and MSDPLB+ policies with multistreaming enabled, the MVG server selects multiple disk volumes for the streams of the client and policy combinations. The streams are distributed with round robin to the list of the primary and historic nodes. It is useful when the data source is large, or a better performance is required. However, it does not contribute to the better deduplication rate.

It is recommended to configure the number of streams as a multiple of the number of disk volumes (for example, 4, 8, 12 for 4 volumes) to ensure even distribution. Any change in stream count or disk volume count affects deduplication in the first backup job after the change.

Assignments can be managed manually to control data distribution and failover behavior. You can predefine primary and historic nodes before a client/policy combination is used.

To manage the assignments of the client and policy combinations with the disk volumes:

1. Get DSID of the MVG volume.

   ```
   /usr/openv/pdde/pdcr/bin/cacontrol –mvg listvvols
   ```

2. Get assigned nodes for a client/policy combination.

   ```
   /usr/openv/pdde/pdcr/bin/cacontrol --cluster get-cp-assignment
   <dsid> [<client> [<policy>]]
   ```

   Output format:

   ```
   <client-name>/<policy-name>: <primary-node>[,<historic-nodes…>]
   ```

   Example:

   ```
   mvgdemo-host1.mvgdemo.com/host4-local-pl01:
   mvgdemo-host4.mvgdemo.com,mvgdemo-host2.mvgdemo.com
   ```

3. Change the primary node of a client/policy combination.

   ```
   /usr/openv/pdde/pdcr/bin/cacontrol --cluster set-cp-assignment
   <dsid> <client> <policy> <msdp-server>
   ```

   Example:

   ```
   cacontrol --cluster get-cp-assignment 3 mvgdemo-host1.mvgdemo.com
    mvg-local-pl11
   mvgdemo-host1.mvgdemo.com/mvg-local-pl11:
   mvgdemo-host2.mvgdemo.com,mvgdemo-host3.mvgdemo.com
   Received 1 client-policy assignments
   ```

   Use the above command to change the primary node that is assigned to a client/policy combination.

   To add a historic node, run the command twice. First, assign the new node as the primary. Then, reassign the original node as the primary. This action moves the newly added node to the historic node list.

   You can also use the command to predefine both primary and historic nodes before the client and policy combination is used. This approach helps you control data distribution and failover behavior more predictably.

The modes of a disk volume in MVG volume

You can set a disk volume in an MVG volume to one of the following modes using the NetBackup web UI.

- Normal

This mode is the default mode. MVG server assigns client and policy combinations to disk volumes in Normal mode when the new jobs run with the MVG volume.

- Read-Only
  MVG server does not assign new client and policy combinations to a disk volume in Read-Only mode. For existing combinations, MVG selects a different disk volume for new backup jobs. This mode is useful when you want to temporarily prevent a disk volume from receiving new backup jobs.
- Maintenance
  Before performing routine server maintenance such as upgrades, patching, or restarts, you should set the corresponding disk volume to Maintenance mode and reset it afterward.
  In this mode, MVG retains existing client and policy combinations on the disk volume, even if the volume becomes temporarily unstable or non-functional. MVG does not assign new combinations to the disk volume during this time.

Recommendation for managing the data distribution

To ensure efficient and balanced data distribution across disk volumes in an MVG volume, follow these recommendations:

- Monitor the trend of disk space usage across all disk volumes in the MVG volume.
- If a disk volume is not utilized well, manually move client/policy assignments to or from it using the `cacontrol --cluster set-cp-assignment` command.
- Add new disk volumes or expand the capacity of existing ones before they become full.
- After adding new disk volumes, manually reassign some client/policy combinations to the new volumes to balance the load between old and new disk volumes.
- Use the node failure handling procedure to manage disk volume failures effectively.
  See “About the node failure management” on page 256.

MVG load balancing levels

MVG server selects a disk volume for a client and policy combination as the primary or historic node based on one of the following load balancing levels when a backup job starts.

By default, MVG uses the Gold level. It is not recommended to use other levels in a customer environment unless necessary.

Table 5-7 MVG load balancing levels

Level	Description
Gold (Default)	The following criteria are used when assigning a new client+policy to a disk volume: ■ Node affinity ■ The used space percentage ■ The evenness of assignment distribution It is used only when the election with node affinity and the used space percentage fails. ■ Source ID hooking Find an available disk volume from the location hashed from the given source ID. Used to find the seed disk volume candidate for used space percentage.
Silver	The following criteria are used when assigning a new client+policy to a disk volume: ■ The used space percentage ■ The evenness of assignment distribution Used only when selection based on used space percentage fails ■ Source ID hooking Find an available disk volume from the location hashed from the given source ID. Used to find the seed disk volume candidate for used space percentage.
Bronze	The following criterion is used when assigning a new client+policy to a disk volume: ■ The evenness of assignment distribution

Table 5-7 MVG load balancing levels *(continued)*

Level	Description
Iron	The following criterion is used when assigning a new client+policy to a disk volume: ■ Source ID hooking Find an available disk volume from the location hashed from the given source ID.

Additional notes:

- In a backup job, with any LB level, whether it is a new or existing client+policy combination, MVG server doesn't return a disk volume that is detected as inactive or full, regardless of the effort it makes. Exceptions include when a disk volume is in maintenance mode, its downtime hasn't exceeded the rebalance freezing time, or in extreme cases where every disk volume in the current MVG volume is unavailable.
- The LB levels do not apply to image read or deletion jobs, as the MVG server directly returns the disk volume that manages the image.
- It is not recommended to use LB levels other than "Gold" in a production environment, unless necessary.
- With the default LB level "Gold", it is recommended to set the media servers of the MSDP servers as the load balance media servers for the MVG server. You can configure this in the NetBackup Web UI under the storage server details.

Specifications and default values of the MVG tuning parameters

The MVG includes a set of configurable tuning parameters that control load balancing, client-policy assignments, cache behavior, and garbage collection. These parameters are typically adjusted for advanced optimization and troubleshooting. Below are the key parameters along with their default values and descriptions.

Table 5-8

Parameter	Default value	Description
`mvg_lb_level`	Gold	MVG load balance level that is used for disk volume selection.
`asmt_rb_freezing_timeout`	1800	Time (in seconds) before reassigning a client+policy from an inactive node.

Table 5-8 *(continued)*

Parameter	Default value	Description
asmt_rb_freezing_on_disk_full	false	When the value set to true, a client and policy combination is not assigned to another node if the current node is full.
asmt_space_imbalance_threshold	15	If a node was initially selected based on an LB level criteria but its space usage exceeds the percentage threshold compared to another node, the other node is selected instead. It is applicable only when the MVG server selects a new node for client+policy.
asmt_scatter_warning	4	The maximum number of nodes a client+policy can be scattered across (excludes MSDPLB+).
asmt_agedout	604800 (7 days)	Maximum duration (in seconds) an assignment can be cached if not on any node.
asmt_max_threshold	3000	Threshold to trigger assignment cleanup.
db_pool_max_size	128	Maximum number of icache DBs in memory pool.
db_obj_timeout	300 (5 minutes)	Time to consider a DB object as recently accessed.
db_agedout	15552000 (180 days)	Maximum age (in seconds) for caching a backup ID in icache DB.
db_prealloc_size	524255 (512 KB)	Preallocated size of icache SQLite DB file.
db_max_size	1048576 (1MB)	Maximum size of icache DB file.
db_short_lifespan	604800 (7 days)	Lifespan threshold for DB records with short remaining life.
gc_interval	604800 (7 days)	Interval for the next round of background GC operation.
gc_start_time	02:20:00 AM	Start time for daily GC operation.

Table 5-8 *(continued)*

Parameter	Default value	Description
`gc_duration`	10800 (3 hours)	Maximum duration for GC operation per day.

CLI for tuning MVG parameters

Starting with NetBackup 11.1, the `mvgapiutil` command-line utility is available to manage MVG tuning parameters. For earlier versions, equivalent REST APIs can be used to achieve the same functionality.

Table 5-9

Task	Command	Equivalent API
Get the load balance configuration.	`mvgapiutil lb getconfig`	`curl -K /uss/config/certificates/controller/.curl_config https://localhost:10100/v1/agent/mvg/loadbalance/config`
Set the load balance configuration (edit).	`mvgapiutil lb setconfig (opens vi editor)`	`curl -K /uss/config/certificates/controller/.curl_config https://localhost:10100/v1/agent/mvg/loadbalance/config -d '{"asmt_rb_freezing_timeout": 3600 }' -X PATCH`
Set the load balance configuration using input file.	`mvgapiutil lb setconfig --input /tmp/mvg-lbconfig.json`	`curl -K /uss/config/certificates/controller/.curl_config https://localhost:10100/v1/agent/mvg/loadbalance/config -d '{"asmt_rb_freezing_timeout": 3600 }' -X PATCH`
Get load balance status (assignments, node state, MVG volume state).	`mvgapiutil lb status`	`curl -K /uss/config/certificates/controller/.curl_config https://localhost:10100/v1/agent/mvg/loadbalance/status`

Example JSON for input file (/tmp/mvg-lbconfig.json):

```
{
  "asmt_rb_freezing_timeout": "15 Minutes",
  "db_agedout": "90 Days"
}
```

Chapter 6

MSDP server evacuation

This chapter includes the following topics:

- About MSDP server evacuation
- Prerequisites for MSDP server evacuation
- Managing the MSDP server evacuation
- Managing the MSDP server evacuation using the command-line
- MSDP server evacuation recommendations
- Migrating the cloud volume of an MVG volume

About MSDP server evacuation

MSDP server evacuation is the process of migrating images from one MSDP server to multiple other MSDP servers. This feature uses MSDP volume group (MVG) to manage the evacuation of an MSDP server. The system remains fully operational during the migration, and no additional changes to the NetBackup configuration are required.

Prerequisites for MSDP server evacuation

Before performing an MSDP server evacuation, ensure that the following prerequisites are met:

- Ensure that the MVG server is in the server list of the primary server in `bp.conf` file. See the following article for more information: https://support.cohesity.com/s/article/article-100017409
- Ensure the following settings on the MVG server:

- NGINX is configured and running on the MVG server. If the MVG server is configured on BYO before NGINX is configured and enabled, run the following command to reconfigure SPWS and NGINX.

  ```
  /usr/openv/pdde/vpfs/bin/vpfs_config.sh --configure_byo
  ```

- Add the MVG evacuation directory in `BPCD_ALLOWED_PATH` in `bp.conf` file.

  ```
  BPCD_ALLOWED_PATH = <mvg-storage-path>/mvg/cache/evac
  ```

  For example, if MVG server is configured in `/msdp` directory,

  ```
  BPCD_ALLOWED_PATH = /msdp/mvg/cache/evac
  ```

- Assign the disk volume of the MSDP server to an MVG volume if it has not been assigned yet. Ensure that the MVG volume includes other disk volumes that are configured in 'Normal' mode. These volumes should have sufficient available space to store the images that are migrated from the MSDP server.
- If the MSDP server contains the existing images that are not created through the MVG volume, take one of the following steps to allow the MVG server to initiate optimized duplication jobs for those images:
  - Option 1: Add all the load balancing media servers of the MVG server in the load balancing media server list of the MSDP server. Alternatively, you can add part of the media servers. By doing so, ensure the storage unit that is used for evacuation has the common media servers specified.
  - Option 2: Configure NetBackup to not require common media servers between the source and the target storage servers by running the following command on the primary or MVG server:

    ```
    /usr/openv/netbackup/bin/admincmd/nbemmcmd -changesetting
    -SHARE_CRED_TO_DUP_READER 1 -machinename <primary server>
    ```

  See "Identifying images not created through the MVG volume on the MSDP server" on page 275.
- If the MSDP server has a cloud volume that needs to be migrated, use the cloud volume migration feature:
  - If the cloud volume is not currently associated with any MVG volume, use the cloud volume migration feature.

    See "About the disaster recovery for cloud LSU" on page 322.
  - If the cloud volume is already part of an MVG volume or you plan to start using it through an MVG volume.

    See "Migrating the cloud volume of an MVG volume" on page 291.
  - If you plan to discontinue the use of the cloud volume and want to migrate all data to other storage locations, you can still use the server evacuation process.

You cannot initiate evacuation of a cloud volume within an MVG volume from the NetBackup web UI. Use MVG CLI commands to start the evacuation. You can then monitor the evacuation status using the CLI or the web UI.

Note: MSDP server evacuation is not supported for any cloud storage types that do not support optimized duplication such as AWS Glacier, AWS Deep Archive, and Microsoft Azure Archive.

If possible, prefer using the cloud volume migration feature over MVG server evacuation for cloud volumes. It is more efficient, as it avoids data movement.

- If a universal share is configured on the MSDP server, perform the universal share backup to the MVG volume. Once the evacuation is complete, you can provision and recover data from the universal share backup.

Identifying images not created through the MVG volume on the MSDP server

You can identify backup images on the MSDP server that are not created through the MVG volume using the NetBackup web UI or the command line.

Identifying images not created through the MVG volume on the MSDP server using web UI

1. On the left, click **Catalog > Search**.
2. In the **Media** section, from the **Disk type** dropdown list, select **PureDisk**.
3. From the **Disk pool** dropdown, select the disk pool of **PureDiskVolum**e of the MSDP server.
4. From the **Volume** dropdown list, select **PureDiskVolume** and set the date and time range, which can cover all the backup images.
5. Click **Search**.

Identifying images not created through the MVG volume on the MSDP server using the command line:

- On the primary server or one of the load balance media servers of the MSDP server, run the following command:

```
/usr/openv/netbackup/bin/admincmd/bpimmedia -d 01/01/2000 -stype
PureDisk -dp <msdp-dp> -dv PureDiskVolume
```

Managing the MSDP server evacuation

Perform the following tasks to manage the MSDP server evacuation process.

Table 6-1

Task	Description
Review the prerequisites.	See "Prerequisites for MSDP server evacuation" on page 273.
Start the MSDP server evacuation.	See "Starting the MSDP server evacuation " on page 276.
Pause or resume the MSDP server evacuation.	See "Pausing or resuming the MSDP server evacuation" on page 277.
Change the destination storage unit for MSDP server evacuation.	See "Changing the destination storage unit for MSDP server evacuation" on page 278.
Monitor the MSDP server evaluation status.	See "Monitoring the MSDP server evaluation status" on page 278.
Handle the failed images gracefully.	See "Handling the failed images gracefully" on page 280.
Clean up the images that are migrated from the MSDP server.	See "Cleaning up the images that are migrated from the MSDP server" on page 281.
Migrate the cloud volume of an MVG volume.	See "Migrating the cloud volume of an MVG volume" on page 291.
Remove the disk volume of the MSDP server from the MVG volume configuration.	See "Removing the disk volume of the MSDP server from the MVG volume configuration" on page 282.
Delete the MSDP server configuration and decommissioning the MSDP/media server.	See "Deleting the MSDP server configuration and decommissioning the MSDP/media server" on page 282.

Starting the MSDP server evacuation

Before initiating the evacuation process for an MSDP server, complete the following preparation steps:

- Review the prerequisites that are required for MSDP server evacuation.
  See "Prerequisites for MSDP server evacuation" on page 273.
- Create a dedicated storage unit that is associated with the current MVG volume you plan to evacuate. This storage unit will be used as the destination storage

of the evacuation. Use a storage unit that is not currently used for regular NetBackup operations to avoid interference during evacuation.

Specify the desired maximum concurrent job number in the storage unit.

See "Configuring the maximum concurrent number of jobs in the storage unit" on page 277.

To start the MSDP server evacuation

1. In the left navigation pane, go to **Storage > Disk storage**.
2. Click the **Disk pools** tab.
3. Locate the MVG disk pool that contains the MVG disk volume with the MSDP server and disk volume to be evacuated.
4. Click the MVG disk pool to open the MVG disk pool details.
5. In the **Disk volumes** table, find the disk volume, and click **Evacuate** from the **Actions** menu on the right.
6. Select the storage unit which is associated with the current MVG volume and click **Start**.

Configuring the maximum concurrent number of jobs in the storage unit

MVG server uses the maximum concurrent job number in the storage unit to determine the maximum number of the optimized duplication jobs for data migration. The evacuation process does not proceed if the number is set to 0. In this case, the state of the disk volume does not change in the **MVG volume details** page.

MVG server gets the number from the latest settings of the storage unit periodically. By default, it is set to every 5 minutes. Run the following MSDP command to change the interval:

```
mvgapiutil evac setconfig
```

This command opens a configuration editor. Set the value for the following parameter:

```
max_cj_refresh_interval
```

Pausing or resuming the MSDP server evacuation

Once the MSDP server evacuation process has started, you can pause or resume it at any time as needed.

To pause or resume the MSDP server evacuation

1. In the left navigation pane, go to **Storage > Disk storage**.
2. Click the **Disk pools** tab.
3. Locate the MVG disk pool that contains the MVG disk volume with the MSDP server and disk volume to be evacuated.
4. Click the MVG disk pool to open the MVG disk pool details.
5. In the **Disk volumes** table, find the disk volume, and click **Pause** or **Resume** from the **Actions** menu on the right.

Changing the destination storage unit for MSDP server evacuation

You can change the destination storage unit for the MSDP server evaluation.

Note: The existing optimized duplication jobs continue to use the previous storage unit. Only the newly created jobs use the new storage unit.

To change the destination storage unit for MSDP server evacuation

1. In the left navigation pane, go to **Storage > Disk storage**.
2. Click the **Disk pools** tab.
3. Locate the MVG disk pool that contains the MVG disk volume with the MSDP server and disk volume to be evacuated.
4. Click the MVG disk pool to open the MVG disk pool details.
5. In the **Disk volumes** table, find the disk volume, and click **Evacuation details** from the **Actions** menu on the right.
6. On the **Evacuation details** page, select a different storage unit from the **Storage Unit** dropdown.

See “Monitoring the MSDP server evaluation status” on page 278.

Monitoring the MSDP server evaluation status

The Evaluation details page of the disk volume contains the following information:

To view the Evacuation details page

1. In the left navigation pane, go to **Storage > Disk storage**.
2. Click the **Disk pools** tab.
3. Locate the MVG disk pool that contains the MVG disk volume with the MSDP server and disk volume to be evacuated.

4 Click the MVG disk pool to open the MVG disk pool details.

5 In the **Disk volumes** table, find the disk volume, and click **Evacuation details** from the **Actions** menu on the right.

Table 6-2 Server evaluation status

Item	Description
Total image count / Total image size	The total number and size of the backup images that are collected from the disk volume of the MSDP server at the beginning of the evacuation process.
Migrated image count / Migrated image size	The total number and size of the backup images that are migrated, including the images, which have and have not been cleaned up from the disk volume of the MSDP server.
Migrated uncleaned image count / Migrated uncleaned image size	The total number and size of the backup images that are migrated but have not been cleaned up from the disk volume of the MSDP server.
In-progress image count / In-progress image size	The total number and size of the backup images that are in the migration process.
Skipped image count / Skipped image size	The total number and size of the backup images that are skipped for evacuation. The orphan images and the failed images whose evacuation is canceled afterwards are counted as skipped images.
Failed image count / Failed image size	The total number and size of the backup images that failed to migrate.
Clean up migrated images	Initiates cleanup of successfully migrated images from the disk volume. If multiperson authorization is enabled, a ticket is generated and must be approved by an authorized administrator. The option Not delete is added in the web UI for the WORM-locked images. If the migrated images are WORM-locked, select the option to initiate the cleanup.

Table 6-2 Server evaluation status *(continued)*

Item	Description
Retry evacuation for all failed images	Adds all images that failed to migrate back to the evacuation backlog to retry migration later.
Cancel evacuation for all failed images	All the images that failed to migrate are skipped from the evacuation.
Images not migrated	Displays a detailed list of backup images that failed to migrate. You can select images from the list to retry the migration or cancel it.

Handling the failed images gracefully

During the evacuation, some images may fail to duplicate for various reasons. You can manage these failures from the **Evacuation Details** page, where you have the option to retry or cancel the evacuation of specific or all failed images.

The following are some common scenarios that cause duplication failures.

Table 6-3 Failed image duplications scenarios

Scenario	Status code	Error messages	Recommended action
The duplication job fails for an SLP-controlled image.	190	`Skipping backup id <image>, it is a lifecycle image that cannot be manually duplicated`	**1** Wait for SLP to process the image or use `nbstlutil cancel` to remove it from SLP operation. **2** Retry from **Evacuation Details** page or MVG CLI.
The image is being duplicated by another SLP job at the same time.	158	`Could not lock backupid: <image>, copy 3; duplicate of image in progress.` `Duplicate of backupid <image> failed, failed accessing daemon lock file (158).`	**1** Wait for the other duplication job to complete. **2** Retry from **Evacuation Details** page or MVG CLI.

Table 6-3 Failed image duplications scenarios *(continued)*

Scenario	Status code	Error messages	Recommended action
Another evacuation process is duplicating an image with the same backup ID.	158 or 226	`Duplicate of backupid <image> failed. The entity already exists (226).`	**1** Wait for current evacuation jobs to complete. **2** Retry from **Evacuation Details** page or MVG CLI. Best practice: evacuate one volume at a time.
The duplication job fails if the image is being expired at the same time.	228, 83, 84, or other	Depends on the timing of image expiration job.	**1** Wait for image expiration job to complete. **2** Retry from **Evacuation Details** page or MVG CLI.
The failure caused by the MSDP server.	N/A	See the job details and the logs of the MSDP server to find more information.	**1** Review job logs and job details. **2** Check MSDP, MVG, and destination MSDP server logs.

Cleaning up the images that are migrated from the MSDP server

You can clean up the images that are migrated from the MSDP server anytime, either during the evacuation process is running, or when all the images have been migrated. It is recommended to do it once they are significantly migrated.

If multiperson authorization is enabled, approve the image cleanup requests by the evacuation process, so that the image cleanup jobs can start. The image cleanup jobs can be found in the Activity Monitor Web UI which cleans up those images from the MSDP server. If some image cleanup jobs fail, ensure that the MSDP server is functional, and the images are allowed to be deleted. For example, unlock the images if they are in WORM locked, or use the **Not delete** option to clean up the images from the NetBackup catalog only.

You can also use the NetBackup command `bpexpdate` to expire the images, which cannot be deleted. After the images are expired, click **Cleanup the migrated images** again to get the status of the images in the MVG server backlog to be refreshed.

To clean up the images that are migrated from the MSDP server

1. In the left navigation pane, go to **Storage > Disk storage**.
2. Click the **Disk pools** tab.

3. Locate the MVG disk pool that contains the MVG disk volume with the MSDP server and disk volume to be evacuated.
4. Click the MVG disk pool to open the MVG disk pool details.
5. In the **Disk volumes** table, find the disk volume, and click **Evacuation details** from the **Actions** menu on the right.
6. Click **Cleanup the migrated images**.

Removing the disk volume of the MSDP server from the MVG volume configuration

When the evacuation details page indicates that all images have been migrated and cleaned up from the MSDP server, and the mode of the disk volume in the MVG volume configuration page shows **Evacuate finished**, you can safely remove the disk volume from the MVG volume configuration.

To clean up the images that are migrated from the MSDP server

1. In the left navigation pane, go to **Storage > Disk storage**.
2. Click the **Disk pools** tab.
3. Locate the MVG disk pool that contains the MVG disk volume with the MSDP server and disk volume to be evacuated.
4. Click the MVG disk pool to open the MVG disk pool details.
5. In the **Disk volumes** table, find the disk volume, and click **Delete** from the **Actions** menu on the right.

If there are any unmigrated or uncleaned images, deleting the disk volume from the MVG volume configuration will result in an error. Wait for all images to finish migrating, then clean up migrated images from the MSDP server. After that, retry or cancel the evacuation of any failed images to ensure that no failed images appear in the evacuation details page.

Deleting the MSDP server configuration and decommissioning the MSDP/media server

Use the instructions in the following articles to delete the MSDP server configuration and to decommission the MSDP server and media server.

- Delete the configuration of the MSDP server: https://support.cohesity.com/s/article/article-100004810
- Decommission the NetBackup media server if it is also a NetBackup media server:

https://support.cohesity.com/s/article/article-100018898

Note: If there are the failed images whose evacuation was canceled, those images may still exist in the NetBackup catalog. In this case, expiring those images is needed. Otherwise, the image expiration step will have no effect as all migrated images have already been cleaned up from the MSDP server.

Managing the MSDP server evacuation using the command-line

The following are the command-line steps to manage the MSDP server evacuation:

Table 6-4 MSDP server evacuation commands

Step	Descritiopn
Start the MSDP server evacuation using the command-line.	See "Starting the MSDP server evacuation using the command-line" on page 284.
Pause the MSDP server evaluation using the command-line.	See "Pausing the MSDP server evacuation using the command-line" on page 285.
Resume the MSEP server evacuation using the command-line.	See "Resuming the MSDP server evacuation using the command-line " on page 286.
Change the destination storage unit using the command-line	See "Changing the destination storage unit using the command-line" on page 287.
Monitor the status of MSDP server evacuation using the command-line.	See "Monitoring the status of MSDP server evacuation using the command-line" on page 288.
Clean up the images that are migrated from the MSDP server using command-line.	See "Cleaning up the images that are migrated from the MSDP server using command-line" on page 289.
Handle the failed images gracefully using the command-line	See "Handling the failed images gracefully using the command-line" on page 290.

Starting the MSDP server evacuation using the command-line

To start the MSDP server evacuation using the command-line

1. Ensure that no storage units exist for the disk volume being evacuated. If any exist, delete them.

   ```
   bpstulist -l

   bpstudel -label <stu>
   ```

2. Create a destination storage unit with the current MVG volume for evacuation.

   ```
   bpstuadd -label <evacuation-stu> -dp <dp-of-mvg-volume> -cj
   <concurrency>
   ```

3. Create a configuration file for evacuation in the following format:

   ```
   V7.5 "operation" "update-virtual-volume" string
   V7.5 "virtualVolume" "<mvg-volume>" string
   V7.5 "diskVolume" "<msdp-server1>:<msdp-volume1>:Y" string
   V7.5 "diskVolume" "<msdp-server2>:<msdp-volume2>:Y" string
   V7.5 "diskVolume" "<msdp-server...>:<msdp-volume...>:Y" string
   ```

   Use `:ES:<storage-unit-of-the-current-mvg-volume>` instead of `:Y` as the suffix for the MSDP server and disk volume, which is to be evacuated.

   For example,

   ```
   # cat /root/evacuate-host4.cfg
   V7.5 "operation" "update-virtual-volume" string
   V7.5 "virtualVolume" "mvg-local" string
   V7.5 "diskVolume" "mvgdemo-host2.mvgdemo.com:PureDiskVolume:Y"
   string
   V7.5 "diskVolume" "mvgdemo-host3.mvgdemo.com:PureDiskVolume:Y"
   string
   V7.5 "diskVolume"
   "mvgdemo-host4.mvgdemo.com:PureDiskVolume:ES:evacuate-stu" string
   ```

4. Apply the configuration changes to the MVG volume.

   ```
   storage_server <mvg-server> -stype PureDisk -configlist
   <configuration-file>
   ```

Pausing the MSDP server evacuation using the command-line

To pause the MSDP server evacuation using the command-line

1 Pause the evacuation instantly.

Update the existing evacuation configuration file or create a new configuration file in the following format:

Use `:EP` instead of `:Y` as the suffix for the MSDP server and disk volume, which is in evacuation mode.

```
V7.5 "operation" "update-virtual-volume" string
V7.5 "virtualVolume" "<mvg-volume>" string
V7.5 "diskVolume" "<msdp-server1>:<msdp-volume1>:Y" string
V7.5 "diskVolume" "<msdp-server2>:<msdp-volume2>:Y" string
V7.5 "diskVolume" "<msdp-server...>:<msdp-volume...>:Y" string
```

For example:

```
cat /root/evacuate-host4.cfg
V7.5 "operation" "update-virtual-volume" string
V7.5 "virtualVolume" "mvg-local" string
V7.5 "diskVolume" "mvgdemo-host2.mvgdemo.com:PureDiskVolume:Y" string
V7.5 "diskVolume" "mvgdemo-host3.mvgdemo.com:PureDiskVolume:Y" string
V7.5 "diskVolume" "mvgdemo-host4.mvgdemo.com:PureDiskVolume:EP" string
```

2 Apply the configuration.

```
nbdevconfig -setconfig -storage_server <mvg-server> -stype
PureDisk -configlist <configuration-file>
```

Alternatively, pause the evacuation by setting the maximum concurrent jobs to 0 in the destination storage unit. The evacuation suspends in up to 5 minutes by default.

Resuming the MSDP server evacuation using the command-line

To resume the MSDP server evacuation using the command-line

1 Update the existing evacuation configuration file or create a new configuration file in the following format:

```
V7.5 "operation" "update-virtual-volume" string
V7.5 "virtualVolume" "<mvg-volume>" string
V7.5 "diskVolume" "<msdp-server1>:<msdp-volume1>:Y" string
V7.5 "diskVolume" "<msdp-server2>:<msdp-volume2>:Y" string
V7.5 "diskVolume" "<msdp-server...>:<msdp-volume...>:Y" string
```

Use `:ES` instead of `:Y` as the suffix for the MSDP server and disk volume, which is in evacuation mode.

For example:

```
cat /root/evacuate-host4.cfg
V7.5 "operation" "update-virtual-volume" string
V7.5 "virtualVolume" "mvg-local" string
V7.5 "diskVolume" "mvgdemo-host2.mvgdemo.com:PureDiskVolume:Y"
string
V7.5 "diskVolume" "mvgdemo-host3.mvgdemo.com:PureDiskVolume:Y"
string
V7.5 "diskVolume" "mvgdemo-host4.mvgdemo.com:PureDiskVolume:ES"
string
```

2 Apply the configuration to resume.

```
nbdevconfig -setconfig -storage_server <mvg-server> -stype
PureDisk -configlist <configuration-file>
```

Changing the destination storage unit using the command-line

To change the destination storage unit using the command-line

1 Update the existing evacuation configuration file or create a new one to specify the destination in the following format:

```
V7.5 "operation" "update-virtual-volume" string
V7.5 "virtualVolume" "<mvg-volume>" string
V7.5 "diskVolume" "<msdp-server1>:<msdp-volume1>:Y" string
V7.5 "diskVolume" "<msdp-server2>:<msdp-volume2>:Y" string
V7.5 "diskVolume" "<msdp-server...>:<msdp-volume...>:Y" string
```

Use `:ES:<storage-unit-of-the-current-mvg-volume>` instead of `:Y` as the suffix for the MSDP server and disk volume, which is to be evacuated.

For example:

```
# cat /root/evacuate-host4.cfg
V7.5 "operation" "update-virtual-volume" string
V7.5 "virtualVolume" "mvg-local" string
V7.5 "diskVolume" "mvgdemo-host2.mvgdemo.com:PureDiskVolume:Y"
string
V7.5 "diskVolume" "mvgdemo-host3.mvgdemo.com:PureDiskVolume:Y"
string
V7.5 "diskVolume"
"mvgdemo-host4.mvgdemo.com:PureDiskVolume:ES:evacuate-stu" string
```

Note: The destination storage unit must be associated with the current MVG volume.

2 Apply the configuration changes to the MVG volume.

```
storage_server <mvg-server> -stype PureDisk -configlist
<configuration-file>
```

Monitoring the status of MSDP server evacuation using the command-line

To monitor the status of MSDP server evacuation using the command-line

- Run the following command to monitor the status.

  ```
  /usr/openv/pdde/pdcr/bin/mvgapiutil evac --mvg-volume <mvg-volume>
  --msdp-server <msdp-server> status
  ```

  For example:

  ```
  /usr/openv/pdde/pdcr/bin/mvgapiutil evac status --mvg-volume
  mvg-local --msdp-server mvgdemo-host4.mvgdemo.com
  {
    "destStorageUnit": "evacuate-stu",
    "failedImages": 0,
    "failedLogicalSize": 0,
    "imagesInProgress": 0,
    "logicalSizeInProgress": 0,
    "migratedImages": 5,
    "migratedImagesUncleaned": 5,
    "migratedLogicalSize": 7246442735,
    "migratedLogicalSizeUncleaned": 7246442735,
    "skippedImages": 0,
    "skippedLogicalSize": 0,
    "state": "the data has been migrated. waiting for cleaning up
  the original images",
    "totalImages": 5,
    "totalLogicalSize": 7246442735
  }
  ```

Parameter	Description
`destStorageUnit`	The target storage unit selected for evacuation.
`totalImages/totalLogicalSize`	The total number and size of the backup images that are collected from the disk volume of the MSDP server at the beginning of the evacuation process.

Parameter	Description
migratedImages/migratedLogicalSize	The total number and size of the backup images that are migrated, including the images, which have and have not been cleaned up from the disk volume of the MSDP server.
migratedImagesUncleaned/migratedLogicalSizeUncleaned	The total number and size of the backup images that are migrated but have not been cleaned up from the disk volume of the MSDP server.
imagesInProgress/logicalSizeInProgress	The total number and size of the backup images that are in the migration process.
skippedImages/skippedLogicalSize	The total number and size of the backup images that are skipped for evacuation. The orphan images and the failed images whose evacuation is canceled afterwards are counted as skipped images.
failedImages/failedLogicalSize	The total number and size of the backup images that failed to migrate.
state	The state of the background evacuation task.

Cleaning up the images that are migrated from the MSDP server using command-line

To clean up the images that are migrated from the MSDP server using command-line

1. List the migrated but uncleaned images.

   ```
   /usr/openv/pdde/pdcr/bin/mvgapiutil evac --mvg-volume <mvg-volume>
   --msdp-server <msdp-server> get-migrated-images-uncleaned
   --no-verify
   ```

2. Expire the images using NetBackup command:

   ```
   bpexpdate -backupid <backup_id> -copy <copy_number> -force -d 0
   -force_not_complete -do_not_follow_dependee [-nodelete]
   [-try_expire_worm_copy]
   ```

   If multiperson authorization (MPA) is enabled, approve the image cleanup requests.

3 Check for remaining uncleaned images with verification:

```
/usr/openv/pdde/pdcr/bin/mvgapiutil evac --mvg-volume <mvg-volume>
--msdp-server <msdp-server> get-migrated-images-uncleaned
```

4 Repeat steps 2 and 4 until no images are listed.

Handling the failed images gracefully using the command-line

Some images may fail to be duplicated for various reasons. When they fail, the status monitoring command shows the number of failed images. Use the following MVG commands to list, retry, or cancel the evacuation of these failed images.

To handle the failed images gracefully using the command-line

1 List the failed images.

```
/usr/openv/pdde/pdcr/bin/mvgapiutil evac --mvg-volume <mvg-volume>
--msdp-server <msdp-server> get-failed-images
```

2 Retry the evacuation of failed images that are specified in the input file.

```
/usr/openv/pdde/pdcr/bin/mvgapiutil evac --mvg-volume <mvg-volume>
--msdp-server <msdp-server> retry-failed-images --input
<input-file>
```

3 Cancel the evacuation of failed images that are specified in the input file:

```
/usr/openv/pdde/pdcr/bin/mvgapiutil evac --mvg-volume <mvg-volume>
--msdp-server <msdp-server> giveup-failed-images --input
<input-file>
```

The input file content is in a JSON format as follows.

```
{
"images": [{ "client": "client", "policy": "policy", "id":
"<backup_id>", "copyNumber": <copy_number>}, ...]
}
```

To apply to all failed images, use an empty list:

```
{
  "images": []
}
```

MSDP server evacuation recommendations

For a smooth and reliable evacuation process for MSDP disk volumes, follow these best practices:

- Set the other disk volumes to read-only mode, which are planned to be evacuated later.
  Before you start evacuation for one disk volume, set the other disk volumes which are planned to be evacuated soon in read-only mode. This step prevents data from being duplicated to those disk volumes.
- Avoid changing volume modes during evacuation if possible.
  If a change is necessary, pause the evacuation, make the change, then resume. This helps reduce temporary failures in ongoing duplication jobs.
- If you have multiple disk volumes to evacuate, evacuate disk volumes one at a time if one of the following conditions is applicable:
  - The MVG server has limited resources (For example, CPU, memory).
    The evacuation backlog is usually small. Once the evacuation is complete and the disk volume is removed from the MVG volume, the backlog is archived on the MVG server.
    If required, you can manually delete the archived evacuation backlog from the MVG server, which is located at:
    `<storage>/mvg/cache/evac/<mvg-volume-dsid>/<msdp-server-of-disk-volume>-archived-<YYYY-mm-DDTHH:MM:SS>`
  - There are duplicated images across volumes. Simultaneous evacuations may try to duplicate different copies of the same image, leading to conflicts and job failures.

Migrating the cloud volume of an MVG volume

If the MSDP server has a cloud volume that needs to be migrated, use the cloud volume migration feature. If the cloud volume is already part of an MVG volume or you plan to start using it through the MVG volume, perform the following steps.

To migrate the cloud volume of an MVG volume

1. Assign the cloud volume to an MVG volume if it is not part of any MVG volume.
2. Set the cloud volume to read-only mode in the MVG volume details page. Delete any storage units that are directly associated with this cloud volume.
3. Migrate images to the MVG Volume. This step is required only if the cloud volume was previously used directly for backups.

You can determine this by checking whether any backup policies or SLPs have written to storage units on this cloud volume. For example, you can query the NetBackup catalog for the images that are associated with the cloud volume.

To migrate the images, run the following commands. Both commands are needed to ensure that the original images can be accessible again from the MVG volume.

- Run the following NetBackup command on the NetBackup primary server:

```
bpimage -newserver "<mvg-server>" -oldserver
"<msdp-server-of-cloud-volume>" -newdiskpool "<mvg-dp>"
-olddiskpool "<cloud-dp>"
```

- Run the following MSDP command on the MSDP server of the cloud volume.

```
cacontrol --dataselection migrate-to-mvg <dsid-of-cloud-volume>
```

4 Delete the cloud volume from the MVG volume configuration without starting an evacuation of the cloud volume.

See "Deleting an MVG volume" on page 245.

5 Remove the cloud volume configuration from NetBackup.

Remove the volume from the old MSDP server or fully decommission the old MSDP server.

See "Deleting a cloud LSU" on page 305.

6 Mount the cloud volume to a new MSDP server and create a new disk pool for the cloud volume.

For information on how to mount and reuse an existing cloud volume to an MSDP server, see the following topic.

See "Reuse cloud LSU" on page 333.

Note: The new MSDP server must not have any other cloud volumes currently used by the MVG volume. This ensures that the cloud volume can be re-added to the MVG volume later.

Run the MSDP command again on the new MSDP server:

```
cacontrol --dataselection migrate-to-mvg <dsid-of-cloud-volume>
```

7 Add the cloud volume back to the MVG volume.

On the **MVG volume details** page, click **Add** to add the cloud volume back to the MVG volume. Alternatively, use the equivalent CLI.

See "Updating an MVG volume using the command-line" on page 243.

Chapter 7

MSDP cloud support

This chapter includes the following topics:

- About MSDP cloud support
- Create a Media Server Deduplication Pool storage server
- Managing credentials for MSDP-C
- Creating a cloud storage unit
- Updating cloud credentials for a cloud LSU
- Updating encryption configurations for a cloud LSU
- Deleting a cloud LSU
- Backup data to cloud by using cloud LSU
- Duplicate data cloud by using cloud LSU
- Configuring AIR to use cloud LSU
- About backward compatibility support
- About the configuration items in cloud.json, contentrouter.cfg, and spa.cfg
- Cloud space reclamation
- About the tool updates for cloud support
- About the disaster recovery for cloud LSU
- About Image Sharing using MSDP cloud
- About restore from a backup in Microsoft Azure Archive
- About FortKnox for NetBackup

- Configuring FortKnox for NetBackup Azure and Azure Government
- Configuring FortKnox for NetBackup Azure and Azure Government using the CLI
- Configuring FortKnox for NetBackup Amazon and Amazon Government
- Configuring FortKnox for NetBackup Amazon and Amazon Government using the CLI
- Configuring FortKnox Google for NetBackup
- Configuring FortKnox Google for NetBackup using the CLI
- Migrating from standard authentication to token-based authentication for Recovery Vault
- About MSDP cloud immutable (WORM) storage support
- About AWS IAM Role Anywhere support
- About Azure service principal support
- About instant access for object storage
- Upgrading to NetBackup 10.3 and cluster environment
- About the cloud direct
- About MSDP lazy delete
- About verification of the images stored in the archive tier

About MSDP cloud support

In this release, NetBackup MSDP cloud support is enhanced to provide a flexible, scalable, high performing, and easy to configure solution, that enables you to leverage cloud storage more efficiently.

Here are the highlights of this feature:

- One MSDP storage server can be configured to support multiple storage targets, including one local storage target and zero or more cloud storage targets. You can move data to local and multiple cloud targets simultaneously.
- The cloud targets can be from the same or different providers, either public, or private. For example, AWS, Azure, HCP, etc.
- The cloud targets can be added on demand after the MSDP server is configured and active.

- Multiple cloud targets can coexist in a single cloud bucket or multiple buckets that are distributed in a single or different cloud providers.
- The data and metadata for local storage and multiple cloud targets are isolated to support Multiple Tenant usage.
- Optimized deduplication is supported within one MSDP server scope so that data can be stored to local storage first and then duplicated to cloud targets in the same media server.
- Disaster recovery from the cloud targets is enhanced and more straightforward.
- Feature is well integrated with the MSDP cluster solution.
- To use universal share or instant access with Cloud LSU, NetBackup Deduplication Engine (spoold) needs more than 0.2% size of MSDP storage capacity to save the fingerprint index files. When you configure universal share or instant access with cloud LSU, ensure that the local drive has enough space to save the fingerprint index files.

Based on the OpenStorage Technology (OST), the new architecture uses multiple Logical storage unit (LSU) to manage and move data. These LSUs can be customized independently to meet different customer requirements. For example, as pure local target (same as MSDP in NetBackup 8.2 or earlier), or local target plus one or more cloud targets.

Starting with NetBackup 8.3, you can configure MSDP from the NetBackup Web UI. You can refer to the NetBackup Web UI documentation for more details.

This chapter focuses on how to use the command line interface to configure MSDP.

Note: To enable OCSD logging information or MSDP cloud, add `loglevel=3` in the section `[Symantec/PureDisk/OpenCloudStorageDaemon]` in `/etc/pdregistry.cfg` on media server and restart the services.

Check the logs at `/<MSDP Storage>/log/ocsd_storage/`.

Operating system requirement for configuration

For the latest and complete information about supported operating systems and platforms, refer to the NetBackup Software Compatibility List (SCL).

Limitations

- Instant access for cloud LSU of AWS Glacier, AWS Deep Archive, and Microsoft Azure Archive is not supported.

- Universal share for cloud LSU of AWS Glacier, AWS Deep Archive, and Microsoft Azure Archive is not supported.
- Accelerator for cloud LSU of AWS Glacier, AWS Deep Archive, and Microsoft Azure Archive is not supported.
- The Cloud LSU for AWS Glacier, AWS Deep Archive, and Microsoft Azure Archive cannot be used as sources of AIR of any types, targeted or classic.
- The Cloud LSU for AWS Glacier, AWS Deep Archive, and Microsoft Azure Archive can be used as targets of optimized duplication but they cannot be used as sources of it.
- Synthetic backup for cloud LSU of AWS Glacier, AWS Deep Archive, and Microsoft Azure Archive is not supported.
- SAP HANA for cloud LSU of Microsoft Azure Archive is not supported.
- Multi-threaded Agent must be disabled when a Client-Direct backup is in use by NetBackup clients that have a NetBackup version earlier than 8.3.
- If you select a load-balancing media server that has NetBackup version earlier than 8.3, then the cloud LSUs are not listed. Even if you select cloud LSUs with a media server that has a NetBackup version earlier than 8.3, the backups can fail.
- Malware scan is not supported on AWS Glacier, AWS Deep Archive, and Microsoft Azure Archive.
- Instant access and universal share features are not supported on SUSE Linux Enterprise.

Create a Media Server Deduplication Pool storage server

Use this procedure to create a Media Server Deduplication Pool storage server. You have the option to create a disk pool (local storage or cloud storage), or an MSDP volume group (MVG), and a storage unit after you create a storage server. The recommendation is that you create the disk pool and storage unit if they do not exist in NetBackup.

To add an MSDP storage server

1. Sign in to the NetBackup web UI.
2. On the left, select **Storage > Disk storage**. Select the **Storage servers** tab, then click **Add**.
3. In the **Storage type** list, select **Disk storage servers**.

4 From the **Category** options, select **Media Server Deduplication Pool (MSDP, MSDP Cloud, MVG)**.

Click **Start**.

5 In **Basic properties**, enter all the required information.

To select your media server, click the search icon. If you do not see the media server you want to use, you can use **Search** field to find it.

Click **Next**.

6 In the **Storage server options**, enter all required information and click **Next**.

If you use Key Management Service (KMS), it must be configured before you can select the **KMS** option.

7 (Optional) In **Media servers**, click **Add** to add any additional media servers that you want to use.

Click **Next**.

8 On the **Review** page, confirm that all options are correct and click **Save**.

If the MSDP storage server creation is unsuccessful, follow the prompts on the screen to correct the issue.

To configure MSDP to use cloud storage, use the following procedure (dropdown in **Volumes** step) to select an existing disk pool volume or create a new one.

9 (Optional) At the top, click on **Create disk pool**.

10 (Optional) To create a cloud logical storage unit and disk pool with replication, click on **Create disk pool**.

Enter the required information to create a disk pool.

In the next tab, select and add the required cloud volume. Select the cloud storage provider and the required details of the storage provider. Enter the credentials to access the cloud storage provider and then define the advanced settings.

For the cloud logical storage unit, click **Edit** to update the **Cloud cache properties** setting in the corresponding disk pool properties page. You must restart the `pdde` services for the updated setting to work.

Additional notes

Review the following additional information:

- Currently, AWS S3 and Azure storage API types are supported.

For more information about the storage API types that NetBackup supports, refer to the topic *About the cloud storage vendors for NetBackup* in the NetBackup Cloud Administrator's Guide.

- When you enable Server-Side Encryption, you can configure AWS Customer-Managed keys. These keys cannot be deleted once they are in use by NetBackup. Each object is encrypted with the key during upload and deleting the key from AWS causes NetBackup restore failures.
- For more information on environments and deployment of FortKnox for NetBackup, refer to the following article:
  https://support.cohesity.com/s/article/article-100051821
  Before you enable the FortKnox Azure and Azure Government options, review the steps from the *Configuring FortKnox Azure and Azure Government* section in the NetBackup Deduplication Guide.
  FortKnox supports multiple options. For FortKnox Azure and Azure Government options in the web UI, you must contact your Cohesity NetBackup account manager for credentials or with any questions.
  Before you enable the FortKnox Amazon and Amazon Government options, review the steps from the *Configuring FortKnox Amazon and Amazon Government* section in the NetBackup Deduplication Guide. FortKnox supports multiple options. For FortKnox Amazon and Amazon Government options in the web UI, you must contact your Cohesity NetBackup account manager for credentials or with any questions.
  Before you enable the FortKnox Google options, review the steps from the *Configuring FortKnox Google for NetBackup* section in the NetBackup Deduplication Guide. For FortKnox Google options in the web UI, you must contact your Cohesity NetBackup account manager for credentials or with any questions.
  See "About FortKnox for NetBackup" on page 362.

Managing credentials for MSDP-C

You can use the Credential management feature to add and edit credentials for MSDP-C that NetBackup uses to connect to a system or workload.

To add credentials to MSDP-C:

1. On the left, click **Credential management**.
2. On the **Named credentials** tab, click **Add**. In the Add credentials dialog, under **Credential store**, select **NetBackup**, and then click **Start**.
3. Under **Basic properties**, provide a **Credential name**. Optionally, provide a **Tag** and **Description**. Click **Next**.

4 Select **MSDP-C** from the **Category** drop-down.

For AWS do the following:

- Select **AWS S3 compatible**.
- If you select **Access key** as the authentication type, specify the **Access key ID** and **Secret access key**. These two parameters authorize NetBackup to access the AWS APIs.
- If you select **IAM Role Anywhere**, do the following:
  - Specify the **Trust Anchor ARN** in the format: `arn:aws:rolesanywhere:region:account:trust-anchor/TA_ID`.
  - Specify the **Profile ARN** in the format: *arn:aws:rolesanywhere:region:account:profile/PROFILE_ID*.
  - Specify the **Role ARN** in the format: *arn:aws::account:role/role-name-with-path*.
  - Enter the **CA Certificate**. The certificate should start with *-----BEGIN CERTIFICATE-----* and end with *-----END CERTIFICATE-----*
  - Enter the **Private Key**. The private key should start with BEGIN RSA PRIVATE KEY----- and end with -----END RSA PRIVATE KEY----- or, it can start with -----BEGIN EC PARAMETERS----- and end with -----END EC PRIVATE KEY-----.

For Azure do the following:

- If you select **Access key** as the authentication type, specify the **Storage account**, and **Access key**.
- If you select **Service principal** as the authentication type, specify the **Storage account**, **Client ID**, **Tenant ID**, and **Secret key**.

5 Click **Next**.

6 Add a role that you want to have access to the credential.

- Click **Add**.
- Select the role.
- Select the credential permissions that you want the role to have.

7 Click **Next**, and follow the prompts to create the credential.

Creating a cloud storage unit

Use the NetBackup web UI or the command line to create a cloud storage unit.

The following steps describe the method to create a cloud storage unit using the command line:

1. Create an MSDP storage server.

   See "Configuring media server deduplication in NetBackup" on page 63.

2. Create a cloud instance alias.

   For example:

   Example 1: Creating an Amazon S3 cloud instance alias

   ```
   # /usr/openv/netbackup/bin/admincmd/csconfig cldinstance -as -in
   amazon.com -sts <storage server> -lsu_name <lsu name>
   ```

   Example 2: Creating an Amazon Glacier cloud instance alias

   ```
   # /usr/openv/netbackup/bin/admincmd/csconfig cldinstance -as -in
   amazon.com -sts <storage server> -lsu_name <lsu name>
   -storage_class GLACIER
   ```

   Example 3: Creating a Microsoft Azure Archive cloud instance alias

   ```
   # /usr/openv/netbackup/bin/admincmd/csconfig cldinstance -as -in
   my -azure -sts <storage server> -lsu_name <lsu name> -storage_tier
   ARCHIVE -post_rehydration_period 3
   ```

   The cloud alias name is *<storage server>_<lsu name>*, and is used to create a bucket.

3. Create a new bucket (Optional)

   For example:

   ```
   # /usr/openv/netbackup/bin/nbcldutil -createbucket -storage_server
   <storage server>_<lsu name> -username <cloud user> -bucket_name
   <bucket name>
   ```

4 Create a configuration file, then run `nbdevconfig` command.

Configuration file content for adding a new cloud LSU:

Configuration setting	Description
`V7.5 "operation" "add-lsu-cloud" string`	Specifies the value "`add-lsu-cloud`" for adding a new cloud LSU.
`V7.5 "lsuName" " " string`	Specifies the LSU name.
`V7.5 "cmsCredName" " " string`	Specifies the credential name.
`V7.5 "lsuCloudBucketName" " " string`	Specifies the cloud bucket name.
`V7.5 "lsuCloudBucketSubName" " " string`	Multiple cloud LSUs can use the same cloud bucket, this value distinguishes different cloud LSUs.
`V7.5 "lsuEncryption" " " string`	Optional value, default is NO. Sets the encryption property for current LSU.
`V7.5 "lsuKmsEnable" " " string`	Optional value, default is NO. Enables KMS for current LSU.
`V7.5 "lsuKmsKeyGroupName" " " string`	Optional value. A Key group name is shared among all LSUs. A Key group name must have valid characters: A-Z, a-z, 0-9, _ (underscore), - (hyphen), : (colon), . (period), and space.
`V7.5 "lsuKmsServerName" " " string`	Optional value. The KMS server name is shared among all LSUs.
`V7.5 "lsuKmsServerType" " " string`	Optional value.
`V7.5 "requestCloudCacheCapacity" "" string`	Optional value. Specifies the disk cache size. If there is no input, a default of 1017 GB is used.

Example 1: Configuration file with encryption disabled

```
V7.5 "operation" "add-lsu-cloud" string
V7.5 "lsuName" "s3amazon1" string
V7.5 "cmsCredName" "cpcp" string
V7.5 "lsuCloudBucketName" "bucket1" string
V7.5 "lsuCloudBucketSubName" "sub1" string
```

Example 2: Configuration file with encryption enabled

```
V7.5 "operation" "add-lsu-cloud" string
V7.5 "lsuName" "s3amazon2" string
V7.5 "cmsCredName" "cpcp" string
V7.5 "lsuCloudBucketName" "bucket1" string
V7.5 "lsuCloudBucketSubName" "sub2" string
V7.5 "lsuEncryption" "YES" string
V7.5 "lsuKmsEnable" "YES" string
V7.5 "lsuKmsKeyGroupName" "test" string
V7.5 "lsuKmsServerName" "test" string
```

Note: All encrypted LSUs in one storage server must use the same `keygroupname` and `kmsservername`. If you use the `nbdevconfig` command to add a new encrypted cloud logical storage unit (LSU) and an encrypted LSU exists in this MSDP, the `keygroupname` must be the same as the `keygroupname` in the previous encrypted LSU.

See "About MSDP Encryption using NetBackup Key Management Server service" on page 114.

Create a configuration file and then run the following `nbdevconfig` command:

```
# /usr/openv/netbackup/bin/admincmd/nbdevconfig -setconfig
-storage_server <storage server> -stype PureDisk -configlist
<configuration file path>
```

Note: The parameter *<storage server>* must be the same as the parameter *<storage server>* in Step 2.

5 Create a disk pool by using the `nbdevconfig` command.

For example:

```
# /usr/openv/netbackup/bin/admincmd/nbdevconfig -previewdv
-storage_servers <storage server name> -stype PureDisk | grep
<LSU name> > /tmp/dvlist

# /usr/openv/netbackup/bin/admincmd/nbdevconfig -createdp -dp
<disk pool name> -stype PureDisk -dvlist /tmp/dvlist
-storage_servers <storage server name>
```

Note: You can also create the disk pool from the NetBackup web UI.

6 Create a storage unit by using `bpstuadd` command.

For example:

```
# /usr/openv/netbackup/bin/admincmd/bpstuadd -label <storage unit
name> -odo 0 -dt 6 -dp <disk pool name> -nodevhost
```

Note: You can also create the storage unit from the NetBackup web UI.

Updating cloud credentials for a cloud LSU

To update cloud credentials for a cloud LSU, you can create a configuration file and then run `nbdevconfig` command.

Configuration file contents for updating cloud credential are as follows:

Configuration setting	Description
`V7.5 "operation" "update-lsu-cloud" string`	Use the value "`update-lsu-cloud`" to update some cloud LSU parameters.
`V7.5 "lsuName" " " string`	Specifies the LSU name.
`V7.5 "cmsCredName" " " string`	Specifies the credential name.

For example:

```
V7.5 "operation" "update-lsu-cloud" string
V7.5 "lsuName" "s3amazon1" string
```

```
V7.5 "cmsCredName" "changedCmsCredName" string
After creating the configuration file, run the following command:

# /usr/openv/netbackup/bin/admincmd/nbdevconfig -setconfig
-storage_server <storage server> -stype PureDisk -configlist
<configuration file path>
```

Note: Use the original storage account to update the FortKnox Azure or FortKnox CMS credentials.

Updating encryption configurations for a cloud LSU

To enable KMS encryption configurations for a Cloud LSU, you can create a configuration file and then run `nbdevconfig` command.

Configuration file contents for updating encryption configurations are as follows:

Configuration setting	Description
`V7.5 "operation" "update-lsu-cloud" string`	You can only update the KMS status from disabled to enabled.
`V7.5 "lsuName" " " string`	Specifies the LSU name.
`V7.5 "lsuKmsEnable" "YES" string`	Specifies the KMS status for the cloud LSU.
`V7.5 "lsuKmsServerName" "" string`	Optional value. KMS server name that is shared among all LSUs.
`V7.5 "lsuKmsKeyGroupName" "" string`	Optional value. Key group name that is shared among all LSUs. Key group name must have valid characters: A-Z, a-z, 0-9, _ (underscore), - (hyphen), : (colon), . (period), and space.

Example to enable KMS status from disabled status to enabled status for cloud LSU "`s3amazon`":

```
V7.5 "operation" "update-lsu-cloud" string
V7.5 "lsuName" "s3amazon" string

V7.5 "lsuKmsEnable" "YES" string
```

```
V7.5 "lsuKmsServerName" "XXX" string
V7.5 "lsuKmsKeyGroupName" "XXX" string
```

Note: All encrypted LSUs in one storage server must use the same `keygroupname` and `kmsservername`. If you use the `nbdevconfig` command to add a new encrypted cloud Logical storage unit (LSU) and an encrypted LSU exists in this MSDP, the `keygroupname` must be the same as the `keygroupname` in the previous encrypted LSU.

For more information, See "About MSDP Encryption using NetBackup Key Management Server service" on page 114.

After creating the configuration file, run the following command:

```
# /usr/openv/netbackup/bin/admincmd/nbdevconfig -setconfig
-storage_server <storage server> -stype PureDisk -configlist
<configuration file path>
```

Deleting a cloud LSU

Use the following steps carefully to remove an MSDP cloud LSU.

If a cloud LSU is associated with an MVG volume, delete the MVG volume first before deleting the cloud LSU. Run the `cacontrol --dataselection list` command to check if the cloud LSU is associated with an MVG volume. If there is an association, the command output displays the cloud LSU description as `UsedByVLSU=<mvg-server>:<mvg-volume>`. For example, in the following output, cloud volume `demohost-cloud` is associated with MVG volume `mvg-cloud` of MVG server `mvgdemo-host1.mvgdemo.com`.

```
cacontrol --dataselection list

[
…
    {
        "dsid": 3,
        "type": 9,
        "name": "demohost-cloud",
        "description":
"UsedByVLSU=mvgdemo-host1.mvgdemo.com:mvg-cloud;"
    }
]
```

See "Deleting an MVG volume" on page 245.

To delete a cloud LSU

1. Expire all images of the cloud LSU in NetBackup.
2. Remove the storage unit and disk pool of this MSDP cloud LSU.
3. Delete all S3 buckets of the cloud LSU if MSDP S3 is configured.
4. To delete a cloud LSU, `storageId` and `CachePath` are needed.

   Run following command to get the information of one cloud LSU:

   ```
   /usr/openv/pdde/pdcr/bin/pddecfg -a listcloudlsu
   dsid, lsuname, storageId, CachePath
   3, S3Volume, server1_ S3Volume/cloud-bucket1/sub1, /msdp/data/ds_3
   4, S3Volume2, server1_ S3Volume2/cloud-bucket1/sub2,
   /msdp/data/ds_4
   ```

   Here the `storageId` of the cloud LSU is "server1_ S3Volume/cloud-bucket1/sub1" and `CachePath` of the cloud LSU is "/msdp/data/ds_3"
5. Run `CRQP` to make sure no `tlog` entries are present in <*msdp_storage_path*>`/spool` folder and <*msdp_storage_path*>`/queue` folder.
6. Delete LSU configurations in `spad` by using `nbdevconfig` command.

   Configuration file contents for deleting an MSDP cloud LSU configuration are as follows:

Configuration setting	Description
`V7.5 "operation" "delete-lsu-cloud" string`	The value "`delete-lsu-cloud`" for deleting the MSDP cloud LSU configurations in `spad`.
`V7.5 "lsuName" " " string`	Specifies the LSU name.

   For example:

   ```
   V7.5 "operation" "delete-lsu-cloud" string
   V7.5 "lsuName" "s3amazon1" string
   ```

   After creating the configuration file, run the following command:

   ```
   # /usr/openv/netbackup/bin/admincmd/nbdevconfig -setconfig
   -storage_server <storage server> -stype PureDisk -configlist
   <configuration file path>
   ```

7. Stop the MSDP service and its monitor service.

```
# /usr/openv/netbackup/bin/nbsvcmon -terminate

# /usr/openv/pdde/pdconfigure/pdde stop
```

8. Delete LSU configurations in `spoold` using the following command:

```
# /usr/openv/pdde/pdcr/bin/spoold --removepartition <storageId>
```

9. Remove the cache and other back-end folders by using the following commands (Optional):

```
# rm -r <CachePath>

# rm -r <msdp_storage_path>/spool/ds_<dsid>

# rm -r <msdp_storage_path>/queue/ds_<dsid>

# rm -r <msdp_storage_path>/processed/ds_<dsid>

# rm -r <msdp_storage_path>/databases/refdb/ds_<dsid>

# rm -r <msdp_storage_path>/databases/datacheck/ds_<dsid>
```

10. Remove the entire sub-bucket folder in cloud. (Optional)

11. Start the MSDP service and its monitor service.

```
# /usr/openv/pdde/pdconfigure/pdde start

# /usr/openv/netbackup/bin/nbsvcmon
```

12. Delete the cloud instance alias.

```
# /usr/openv/netbackup/bin/admincmd/csconfig cldinstance -rs -in
<instance_name> -sts <storage_server_name>_<lsu_name>
```

Backup data to cloud by using cloud LSU

Run the following steps to backup data to cloud LSU:

- Create a cloud LSU and related disk pool and storage unit (cloud storage unit).
- Create a backup policy and use cloud storage unit as policy storage.
- Run the backup and the data is written to cloud storage.

You can create and back up to multiple Cloud LSUs at the same storage server.

Duplicate data cloud by using cloud LSU

Run the following steps to duplicate backup images from local MSDP to cloud LSU:

- Configure an MSDP storage server and create a disk pool by using "`PureDiskVolume`" and then create a storage unit (local storage unit).
- Create a cloud LSU and related disk pool and storage unit (cloud storage unit).
- Create a Storage Lifecycle Policy and add "Backup" and "Duplication" values. Data is backs up to local storage unit and then duplicates to a cloud storage unit.
- Create a backup policy and use storage lifecycle policy as policy storage.
- Run the backup and the data is written to local storage and then duplicated to cloud storage.

Duplication can also be done from cloud LSU to local MSDP and between two cloud LSUs.

Configuring AIR to use cloud LSU

The following steps describe the tasks that are required to replicate the backup images from one LSU to another in a different NetBackup domain:

- See "Configuring MSDP replication to a different NetBackup domain" on page 137.
- Configure a trust relationship with the target NetBackup domain.
  See "About trusted primary servers for Auto Image Replication" on page 146.
- Add an LSU in the remote storage server as a replication target.
  To add a replication target in a different NetBackup domain, you can use NetBackup Web UI or use the command line interface.

1. Create a configuration file for adding a replication target.

   Configuration file content for adding a replication target:

Configuration setting	Description
`V7.5 "operation" " " string`	The value must be “set-replication” for adding a new replication target.
`V7.5 " rephostname" " " string`	Specifies the replication target's host name.
`V7.5 "replogin" " " string`	Specifies the replication target storage server's user name.
`V7.5 "reppasswd" " " string`	Specifies the replication target storage server's password.
`V7.5 "repsourcevolume" " " string`	Specifies the replication source volume name.
`V7.5 "reptargetvolume" " " string`	Specifies the replication target volume name.

   Example:

```
[root@sourceserver~]# cat add-replication-local2cloud.txt
V7.5 "operation" "set-replication" string
V7.5 "rephostname" "targetserver1.example.com" string
V7.5 "replogin" "root" string
V7.5 "reppasswd" "root" string
V7.5 "repsourcevolume" "PureDiskVolume" string
V7.5 "reptargetvolume" "s3amamzon1" string
```

   After creating the configuration file, run the `nbdevconfig` command.

   For example:

```
# /usr/openv/netbackup/bin/admincmd/nbdevconfig -setconfig
-storage_server <storage server> -stype PureDisk -configlist
<configuration file path>
```

2. Run `nbdevconfig` to update the disk volume.

   For example:

```
# /usr/openv/netbackup/bin/admincmd/nbdevconfig -updatedv -stype
PureDisk -dp diskpool1 -media_server sourceserver.example.com
```

- Configure a storage lifecycle policy.
  You must create an import SLP in the target domain before you configure an SLP in the source domain.
  See “About storage lifecycle policies” on page 162.

See “About the storage lifecycle policies required for Auto Image Replication” on page 163.
See “Creating a storage lifecycle policy” on page 165.

Removing a replication target

Complete the following steps to delete a replication target:

1. Create a configuration file for deleting a replication target.

   Configuration file content for deleting a replication target:

Configurtion setting	Description
`V7.5 "operation" " " string`	The value must be “delete-replication” for deleting a new replication target.
`V7.5 " rephostname" " " string`	Specifies the replication target's host name.
`V7.5 "repsourcevolume" " " string`	Specifies the replication source volume name.
`V7.5 "reptargetvolume" " " string`	Specifies the replication target volume name.

   For example:

```
[root@sourceserver~]# cat delete-replication-local2cloud.txt
V7.5 "operation" "delete-replication" string
V7.5 "rephostname" "targetserver1.example.com" string
V7.5 "repsourcevolume" "PureDiskVolume" string
V7.5 "reptargetvolume" "s3amamzon1" string
```

   After creating the configuration file, run the `nbdevconfig` command.

   For example:

```
# /usr/openv/netbackup/bin/admincmd/nbdevconfig -setconfig
-storage_server <storage server> -stype PureDisk -configlist
<configuration file path>
```

2. Run `nbdevconfig` to update the disk volume.

   For example:

```
# /usr/openv/netbackup/bin/admincmd/nbdevconfig -updatedv -stype
PureDisk -dp diskpool1 -media_server sourceserver.example.com
```

About backward compatibility support

To replicate an image from an earlier version (NetBackup 8.2 or earlier) of an MSDP server to the cloud LSU of target MSDP server, you need a username with the cloud LSU name when you add the A.I.R. target. Use the web UI to add an A.I.R. target. The format of the username with the target cloud LSU is:

```
<username>?LSU=<target cloud LSU>
```

For example, there is a target storage server and the username of the server is `userA` and there is a cloud LSU `s3cloud1` in the target storage server. To replicate an image from an old storage server to the cloud LSU of the target server, you can use the following username while adding the A.I.R. target:

```
userA?LSU=s3cloud1
```

You must also create an import SLP for the local volume of the target storage server in the target primary server. Then select the imported SLP when you create the target A.I.R. SLP on the source side. When the A.I.R. runs, the import job on the target side shows the policy name as **SLP_No_Target_SLP** in the Activity monitor, but the data is sent to cloud.

If the NetBackup client version is 8.2 or earlier, the client direct backup from the old client to cloud LSU of one storage server might fail. During the backup if `mtstrmd` is used on the client side, the job fails with a media write error. To disable `mtstrmd` at the client side, open the configuration file `pd.conf` on the client and change the following:

`MTSTRM_BACKUP_ENABLED = 1` to `MTSTRM_BACKUP_ENABLED = 0`.

The `pd.conf` file is located in the following directories:

- UNIX

  `/usr/openv/lib/ost-plugins/`

- Windows

  install_path`\Veritas\NetBackup\bin\ost-plugins`

When a client direct backup runs with a cloud LSU and an old client, the client does only client-side deduplication.

To use cloud LSU, the load balance server of the storage server must not be an earlier version (NetBackup 8.2 or earlier). If there are new and old load balancers, a new load balance server is selected automatically to make sure that the job can be done successfully. When you restore a backup image on cloud LSU and you select the media server explicitly, the media server that you select must not be an earlier version of NetBackup.

About the configuration items in cloud.json, contentrouter.cfg, and spa.cfg

The `cloud.json` file is available at: `<STORAGE>/etc/puredisk/cloud.json`.

The file has the following parameters:

Parameter	Details	Default value
`UseMemForUpload`	If it is set to true, the upload cache directory is mounted in memory as `tmpfs`. It is especially useful for high speed cloud that disk speed is bottleneck. It can also reduce the disk competition with local LSU. The value is set to true if the system memory is enough. The default value is true if there is enough memory available.	true
`CachePath`	The path of the cache. It is created under an MSDP volume according to the space usage of MSDP volumes. It will reserve some space that local LSU cannot write beyond. Usually you do not need to change this path, unless in some case that some volumes are much freer than others, multiple cloud LSUs may be distributed to the same disk volume. For performance consideration, you may need to change this option to make them distributed to different volumes. This path can be changed to reside in a non-MSDP volume.	NA
`UploadCacheGB`	It is the maximum space usage of upload cache. Upload cache is a subdirectory named “upload” under `CachePath`. For performance consideration, it should be set to larger than: (max concurrent write stream number) * `MaxFileSizeMB` * 2. So, for 100 concurrent streams, about 13 GB is enough. **Note:** The initial value of `UploadCacheGB` in the `cloud.json` file is the value of `CloudUploadCacheSize` in the `contentrouter.cfg` file. When you add a new cloud LSU, the value of `UploadCacheGB` is equal to `CloudUploadCacheSize`. You can later change this value in the `cloud.json` file.	12

Parameter	Details	Default value
DownloadDataCacheGB	It is the maximum space usage of data file, mainly the SO BIN file. The larger this cache, the more data files can reside in the cache. Then there is no need to download these files from cloud when doing restore. **Note:** The initial value of DownloadDataCacheGB in the cloud.json file is the value of CloudDataCacheSize in the contentrouter.cfg file. When you add a new cloud LSU, the value of DownloadDataCacheGB is equal to CloudDataCacheSize. You can later change this value in the cloud.json file.	500
DownloadMetaCacheGB	It is the maximum space usage of metadata file, mainly the DO file and SO BHD file. The larger this cache, the more meta files can reside in the cache. Then there is no need to download these files from cloud when doing restore. **Note:** The initial value of DownloadMetaCacheGB in the cloud.json file is the value of CloudMetaCacheSize in the contentrouter.cfg file. When you add a new cloud LSU, the value of DownloadMetaCacheGB is equal to CloudMetaCacheSize. You can later change this value in the cloud.json file.	500
MapCacheGB	It is the max space usage of map file that is used for compatibility of MD5 type fingerprint. The larger this cache, the more map files can reside in the cache. **Note:** The initial value of MapCacheGB in the cloud.json file is the value of CloudMapCacheSize in the contentrouter.cfg file. When you add a new cloud LSU, the value of MapCacheGB is equal to CloudMapCacheSize. You can later change this value in the cloud.json file.	5
UploadConnNum	Maximum number of concurrent connections to the cloud provider for uploading. Increasing this value is helpful especially for high latency network.	60
DataDownloadConnNum	Maximum number of concurrent connections to the cloud provider for downloading data. Increasing this value is helpful especially for high latency network.	40

Parameter	Details	Default value
MetaDownloadConnNum	Maximum number of concurrent connections to the cloud provider for downloading metadata. Increasing this value is helpful especially for high latency network.	40
MapConnNum	Maximum number of concurrent connections to the cloud provider for downloading map.	40
DeleteConnNum	Maximum number of concurrent connections to the cloud provider for deleting. Increasing this value is helpful especially for high latency network.	100
KeepData	Keep uploaded data to data cache. The value always false if UseMem is true.	false
KeepMeta	Keep uploaded meta to meta cache, always false if UseMem is true.	false
ReadOnly	LSU is read only, cannot write and delete on this LSU.	false
MaxFileSizeMB	Max size of bin file in MB.	64
WriteThreadNum	The number of threads for writing data to the data container in parallel that can improve the performance of IO.	2
RebaseThresholdMB	Rebasing threshold (MB), when image data in container less than the threshold, all of the image data in this container will not be used for deduplication to achieve good locality. Allowed values: 0 to half of MaxFileSizeMB, 0 = disabled	4
AgingCheckContainerIntervalDay	The interval of checking a container for this Cloud LSU (in days). **Note:** For upgraded system, you must add this manually if you want to change the value for a cloud LSU.	180

The contentrouter.cfg file is available at:
<*STORAGE*>/etc/puredisk/contentrouter.cfg.

The file has the following parameters:

Parameter	Details	Default value
CloudDataCacheSize	Default data cache size when adding Cloud LSU. Decrease this value if enough free space is not available.	500 GiB
CloudMapCacheSize	Default map cache size when adding Cloud LSU. Decrease this value if enough free space is not available.	5 GiB

Parameter	Details	Default value
CloudMetaCacheSize	Default meta cache size when adding Cloud LSU. Decrease this value if enough free space is not available.	500 GiB
CloudUploadCacheSize	Default upload cache size when adding Cloud LSU. The minimum value is 12 GiB.	12 GiB
MaxPredictiveCacheSize	Specify the maximum predictive cache size. It is based on total system memory, swap space excluded.	20 %
CloudBits	The number of top-level entries in the cloud cache. This number is (2^CloudBits). Increasing this value improves cache performance, at the expense of extra memory usage. Minimum value = 16, maximum value = 48.	Auto-sized according to MaxCloudCacheSize
DCSCANDownloadTmpPath	While using the dcscan to check cloud LSU, data gets downloaded to this folder. For details, see the dcscan tool in cloud support section.	disabled
UsableMemoryLimit	Specify the maximum usable memory size in percentage. MaxCacheSize + MaxPredictiveCacheSize + MaxSamplingCacheSize + Cloud in-memory upload cache size must be less than or equal to the value of UsableMemoryLimit	85%
MaxSamplingCacheSize	Specify the maximum sampling cache size in percentage for all LSUs here. If you want to limit the maximum sampling cache size for a cloud LSU, you can configure LSUSamplingCachePercent in cloud.json. The default value of this parameter is -1.0% which means no limitation. Sampling cache is also used to implement global deduplication for MSDP AKS and MSDP FlexScale clusters.	5%
ClusterHookEngineCount	Global deduplication uses history data to optimize sampling cache hookup process. When the history data is valid, only remote s-cache lookup request is sent to the number of ClusterHookEngineCount nodes to reduce the cross-node overheads. To disable this feature, set ClusterHookEngineCount to 0.	3
ClusterHookMinHistoryAgeInSecond	The minimum age in seconds for the history data to be valid. The data newer than the minimum age is not used.	604800

Parameter	Details	Default value
`ClusterHookMaxHistoryAgeInSecond`	The maximum age in seconds for the valid history data. The data older than the maximum age is removed.	2592000

Adding a new cloud LSU fails if no partition has free space more than the following:

```
CloudDataCacheSize + CloudMapCacheSize + CloudMetaCacheSize +
CloudUploadCacheSize + WarningSpaceThreshold * partition size
```

Use the `crcontrol --dsstat 2` command to check the space of each of the partition.

Note: Each Cloud LSU has a cache directory. The directory is created under an MSDP volume that is selected according to the disk space usage of all the MSDP volumes. Cloud LSU reserves some disk space for cache from that volume, and the local LSU cannot utilize more disk space.

The initial reserved disk space for each of the cloud LSU is the sum of values of `UploadCacheGB`, `DownloadDataCacheGB`, `DownloadMetaCacheGB`, and `MapCacheGB` in the `<STORAGE>/etc/puredisk/cloud.json` file. The disk space decreases when the caches are used.

There is a `Cache` options in `crcontrol --dsstat 2` output:

```
# crcontrol --dsstat 2

=============== Mount point 2 ===============

Path = /msdp/data/dp1/1pdvol

Data storage

Raw Size Used Avail Cache Use%

48.8T 46.8T 861.4G 46.0T 143.5G 2%

Number of containers : 3609

Average container size : 252685915 bytes (240.98MB)

Space allocated for containers : 911943468161 bytes (849.31GB)

Reserved space : 2156777086976 bytes (1.96TB)

Reserved space percentage : 4.0%
```

The `Cache` option is the currently reserved disk space by cloud for this volume. The disk space is the sum of the reserved space for all cloud LSUs that have cache directories on this volume. The actually available space for Local LSU on this volume is `Avail - Cache`.

The `spa.cfg` file is available at: `<STORAGE>/etc/puredisk/spa.cfg`.

The file has the following parameters:

Parameter	Details	Default value
`CloudLSUCheckInterval`	The check cloud LSU status interval in seconds.	1800
`EnablePOIDListCache`	The status of the POID (Path Object ID) list cache as enabled or disabled. Path Object contains the metadata associated with that image. .	true

Cloud space reclamation

MSDP stores data segments in data containers, and data containers are sent to cloud storage and stored as objects. The segments in one container may belong to the different backup images. When backup images expire, their segments become garbage. If all segments in one container are garbage, the whole container can be reclaimed. If not, that container cannot be reclaimed because it contains both useful data and garbage. A small number of segments in one container may prevent that container from being reclaimed for a long time if those segments are referenced by many backup images.

MSDP uses container aging and cloud compaction to reclaim the space in these containers.

Configuring the container aging

Container aging tries to identify the containers that contain a small number of segments but live for a long time, and exclude them from being referenced by new backup images.

The `contentrouter.cfg` file has the following aging check related parameters:

Parameter	Description	Default value
`EnableAgingCheck`	Enable or disable Cloud LSU container aging check.	true
`AgingCheckAllContainers`	This parameter determines whether to check all containers or not. If set to 'false', it only checks containers in some latest images.	false
`AgingCheckSleepSeconds`	Aging check thread wakes up periodically with this time interval (in seconds).	20
`AgingCheckBatchNum`	The number of containers for aging check each time.	400
`AgingCheckContainerInterval`	Default interval value of checking a container when adding Cloud LSU (in days).	180
`AgingCheckSizeLowBound`	This threshold is used to filter the containers whose size is less than this value for aging check.	8Mib
`AgingCheckLowThreshold`	This threshold is used to filter the containers whose garbage percentage is less than this value (in percentage).	10%

After you update the aging check related parameters, you must restart the MSDP service. You can use the **crcontrol** command line to update those parameters without restarting MSDP service.

To update the aging parameters using crcontrol command line

1. Enable cloud aging check for all cloud LSUs.

   ```
   /usr/openv/pdde/pdcr/bin/crcontrol --cloudagingcheckon
   ```

2. Enable cloud aging check for a specified cloud LSU.

   ```
   /usr/openv/pdde/pdcr/bin/crcontrol --cloudagingcheckon <dsid>
   ```

3. Disable cloud aging check for all cloud LSUs.

   ```
   /usr/openv/pdde/pdcr/bin/crcontrol --cloudagingcheckoff
   ```

4. Disable cloud aging check for a specified cloud LSU.

   ```
   /usr/openv/pdde/pdcr/bin/crcontrol --cloudagingcheckoff <dsid>
   ```

5. Show cloud aging check state for all cloud LSUs.

   ```
   /usr/openv/pdde/pdcr/bin/crcontrol --cloudagingcheckstate
   ```

6. Show cloud aging check state for a specified cloud LSU.

   /usr/o`penv/pdde/pdcr/bin/crcontrol --cloudagingcheckstate <dsid>`

7. Change cloud aging check to fast mode for all cloud LSUs.

   ```
   /usr/openv/pdde/pdcr/bin/crcontrol --cloudagingfastcheck
   ```

8. Change cloud aging check to fast mode for a specified cloud LSU.

   ```
   /usr/openv/pdde/pdcr/bin/crcontrol --cloudagingfastcheck <dsid>
   ```

Configuring the cloud compaction

Cloud compaction replaces the existing container with the garbage with a new smaller container without the garbage. It is similar to compaction for local storage. In the compaction process, containers that meet the conditions are downloaded from object cloud storage to local copies. Compaction removes garbage in the local copies and uploads them to overwrite the old objects in the cloud. In this process, data is transferred to and from cloud storage and may incur extra cost. It also needs reasonable bandwidth for compaction to work efficiently. Hence, cloud compaction is disabled by default. It can be enabled in the deployment where cost and bandwidth are not a problem. For example, it's recommended to enable it in a private cloud or in case MSDP is deployed in the same cloud as object storage.

Limitations:

- WORM cloud LSU is not supported.
- Buckets with versioning enabled are not supported.
- CloudCatalyst migration system is not supported.

- MSDP Cluster deployments (AKS, EKS, NBFS) are not supported.

You can configure cloud compaction with the following options in `cloud.json`

Parameter	Description	Default value
`CompactEnable`	Enable or disable cloud compaction.	false
`CompactBatchNum`	The number of containers for cloud compaction each time.	400
`EnableCompactCandidateCheck`	Enable or disable cloud compaction check. This value is available when `CompactEnable` is true.	true
`CompactLboundMB`	Filter the containers that have garbage size less than this value for cloud compaction.	16
`CompactSizeLboundMB`	Filter the containers with the size less than this value for cloud compaction.	32
`CompactMaxPo`	Filter the containers that are referenced more than this number of path objects for cloud compaction.	100

To update the cloud compaction parameters using crcontrol command line

1. Enable cloud compaction for one cloud LSU.

   `/usr/openv/pdde/pdcr/bin/crcontrol --cloudcompacton <dsid>`

   This command enables compaction temporarily. To enable compaction permanently, update `CompactEnable` value manually in `cloud.json`.

   Container aging is disabled automatically when cloud compaction is enabled.

2. Disable cloud compaction for one cloud LSU.

   `/usr/openv/pdde/pdcr/bin/crcontrol --cloudcompactoff <dsid>`

   To disable compaction permanently, update `CompactEnable` manually in `cloud.json`, and restart spoold.

3. Show cloud compaction state for all cloud LSUs.

   `/usr/openv/pdde/pdcr/bin/crcontrol --cloudcompactstate`

About the tool updates for cloud support

`DCSCAN`:

`Dcscan` downloads data container from the cloud. The default download path is `<STORAGE>/tmp/DSID_#dsid`, where `#dsid` is dependent on the cloud LSU DSID value. Different cloud storage providers have different DSID values. You do not

need to know the DSID value, dcscan gets the DSID value automatically. The default download path can be modified in the contentrouter.cfg file using the DCSCANDownloadTmpPath field.

While using the dcscan tool to look at cloud data, -a option is disabled, because it downloads all data containers from cloud, it is an expensive operation. The -fixdo option is disabled as well, because dcscan only downloads data container from the cloud. Others operations are same as the local LSU.

dcscan downloads data containers to its own cache. When compaction is enabled for some LSU, before running dcscan for this LSU, remove those stale containers from dcscan cache directory.

SEEDUTIL:

Seedutil can be used for seeding a backup for a better deduplication rate. It creates links in the <destination client name> directory to all the backup files found in the path <client name>/<policy name> that have <backup ID> in their names. The user needs to know which DSID value the cloud LSU has used. That DSID value needs to be given to the seedutil, to let seedutil know which cloud LSU will seed a client. If you do a seeding for a local LSU, the default DSID is 2, you do not need to give the DSID value. Seedutil cannot seed across different DSIDs.

For example,/usr/openv/pdde/pdag/bin/seedutil -seed -sclient <*source_client_name*> -spolicy <*source_policy_name*> -dclient <*destination_client_name*> -dsid <*dsid_value*>.

CRCONTROL

Using crcontrol –clouddsstat option to show cloud LSU datastore usage. DSID value needs to be given. As cloud storage has unlimited space, the size is hard-coded to 8 PB.

For example:

```
# /user/openv/pdde/pdcr/bin/crcontrol --clouddsstat <dsid_value>
************ Data Store statistics ************
Data storage      Raw    Size   Used   Avail  Use%
8.0P   8.0P  80.9G  8.0P   0%
Number of containers             : 3275
Average container size           : 26524635 bytes (25.30MB)
Space allocated for containers   : 86868179808 bytes (80.90GB)
Reserved space                   : 0 bytes (0.00B)
Reserved space percentage        : 0.0%
```

CRSTATS:

Using `crstats -cloud -dsid` option to show the cloud LSU statistics. `DSID` value needs to be given. As cloud storage has unlimited space, the size is hard-coded to 8 PB.

For example:

```
#/usr/openv/pdde/pdcr/bin/crstats --cloud-dsid <dsid_value>
Storage Pool Raw Size=9007199254740992Bytes
Storage Pool Reserved Space=0Bytes
Storage Pool Required Space=0Bytes
Storage Pool Size=9007199254740992Bytes
Storage Pool Used Space=86868179808Bytes
Storage Pool Available Space=9007112386561184Bytes
Catalog Logical Size=402826059439Bytes
Catalog files Count=3726
Space Allocated For Containers=86868179808Bytes
Deduplication Ratio=4.6
```

`PDDECFG`:

Using `pddecfg` to list all the cloud LSUs.

For example:

```
/usr/openv/pdde/pdcr/bin/pddecfg -a listcloudlsu
dsid, lsuname, storageId, CachePath
3, S3Volume, amazon_1/cloud-bucket1/sub1, /msdp/data/ds_3
4, S3Volume2, amazon_1/cloud-bucket1/sub2, /msdp/data/ds_4
```

About the disaster recovery for cloud LSU

If the disk on which the NetBackup software resides or the disk on which the deduplicated data resides fails, you can use the following steps to recover the system and data depending on different scenarios.

After recovery, your NetBackup deduplication environment functions normally. Any valid backup images on the cloud LSU storage are available for restores.

Before you start disaster recovery, ensure that:

- Media server on which the MSDP service resides still works. If media server does not work, you must reinstall the media server. Refer to the *NetBackup Installation Guide* for reinstalling media server software.
- KMS server is ready if KMS encryption is used by cloud LSU.

- If you reuse a cloud LSU in a different domain, ensure that you add the correct cloud credentials with the same name using the Credential Management in web UI.

After disaster recovery for cloud LSU, importing backup images is needed with following cases:

- The primary server has no catalog of images in MSDP storage. For example, when the primary server is reinstalled and catalog in the primary is lost. The catalog is needed to import backup images. Refer section “About importing backup images” in *NetBackup Administrator's Guide, Volume I* for more information.
- The primary server has an incorrect catalog record about the MSDP storage. The catalog in the primary server is not correct any more as the storage server is moved to a new media server after the disaster recovery. To correct the catalog on the primary server, run the `bpimage` command. The new media server here means a newly added media server or other existing media server.
- When the primary has catalog of images in MSDP storage and the same media server is used to do disaster recovery, it’s not needed to do backup images importing.
- Imports are not supported for backups stored on Amazon S3 Glacier, Deep Archive, and Microsoft Azure Archive cloud LSUs if they were previously migrated from Cloud Catalyst.

You can do the disaster recovery for cloud LSU with the following three steps:

1. Set up the MSDP storage server with local storage.
2. Add a cloud LSU to reuse existing cloud data.
3. Perform backup images importing if the catalog is not available in the primary server.

Scenario 1: The MSDP server name has not changed, the local storage is lost, and the NetBackup catalog is present

Step	Task	Procedure
1	Create an empty local LSU	See “Configure/Reconfigure MSDP local storage” on page 331.
2	Reuse Cloud LSU	See “Reuse cloud LSU” on page 333.

Scenario 2: The MSDP server name has not changed, the local storage is lost, and the NetBackup catalog is not present

Step	Task	Procedure
1	Delete old storage server-related configurations	Perform the following steps from *Recovering from an MSDP storage server failure* topic.See "Recovering from an MSDP storage server failure" on page 584. ■ Delete the storage units that use the disk pool. ■ Delete the disk pool. ■ Delete the deduplication storage server. ■ Delete the deduplication host configuration file. ■ Delete the credentials on deduplication servers.
2	Configure the new storage server.	Run the following command on the primary server: `/usr/openv/netbackup/bin/admincmd/nbdevconfig -creatests -storage_server "storage server" -stype PureDisk -media_server "media server" -st 9` **Note:** You can also create the storage server from the NetBackup web UI.
3	Create an empty local LSU.	See "Configure/Reconfigure MSDP local storage" on page 331. **Note:** Skip these steps if the storage server was created using the NetBackup web UI.
4	Reuse cloud LSU.	See "Reuse cloud LSU" on page 333.
5	Create disk pool for cloud LSUs.	Run the following commands on the primary server: `/usr/openv/netbackup/bin/admincmd/nbdevconfig -previewdv -storage_servers <storage server name> -stype PureDisk | grep <LSU name> > /tmp/dvlist` `/usr/openv/netbackup/bin/admincmd/nbdevconfig -createdp -dp <disk pool name> -stype PureDisk -dvlist /tmp/dvlist -storage_servers <storage server name>` **Note:** You can also create the disk pool from the NetBackup web UI.

Step	Task	Procedure
6	Prepare images for import on high-latency storage.	If the source version is 11.0 or later, MSDP automatically prepares images for import in the background. Wait for images to be ready for the import. Run the following command to check the status: `/usr/openv/pdde/pdcr/bin/tiermover --status --lsu <LSU name> [--client <client>] [--policy <policy>] [--backupid <backup ID>] [--sobins] [--active] [--lsulist] [--debug] [--verbose]` If the source version is earlier than 11.0 or the source server is down and did not finish pre-import operations, run the following command to prepare the images for import: `/usr/openv/pdde/pdcr/bin/tiermover --start --lsu <LSU name> --client <client> --policy <policy> [--backupid <backup ID>] [--retrieval Bulk|Standard|Expedited] [--verbose] [--debug]` **Note:** Import fails if the images are not ready for the import.
7	Import the images back.	Do two-phase import. See NetBackup Administrator's Guide, Volume I For the high-latency storage when the source is 11.0 or later: If import job for high-latency storage fails, it may be because the source storage did not finish preparing the images for the import before the disaster occurred. In that case, perform the step 6 again to prepare the images for import.

Scenario 3: The MSDP server name has changed, the local storage is not lost, and the NetBackup catalog is present

Step	Task	Procedure
1	Ensure that the old MSDP server is stopped if recovering from an old MSDP server to a new MSDP server.	Run the following commands on the old MSDP server: `/usr/openv/netbackup/bin/bp.kill_all`
2	Configure the new storage server.	Run the following command on the primary server: `/usr/openv/netbackup/bin/admincmd/nbdevconfig -creatests -storage_server "storage server" -stype PureDisk -media_server "media server" -st 9` **Note:** You can also create the storage server from the NetBackup web UI.

Step	Task	Procedure
3	Reuse an existing local storage path.	See "Configure/Reconfigure MSDP local storage" on page 331. **Note:** Skip these steps if the storage server was created using the NetBackup web UI.
4	Restart the storage server.	Run the following commands on the primary server: `/usr/openv/netbackup/bin/bp.kill_all` `/usr/openv/netbackup/bin/bp.start_all`
5	Create disk pool for cloud LSUs.	Run the following commands on the primary server: `/usr/openv/netbackup/bin/admincmd/nbdevconfig -previewdv -storage_servers <storage server name> -stype PureDisk | grep <LSU name> > /tmp/dvlist` `/usr/openv/netbackup/bin/admincmd/nbdevconfig -createdp -dp <disk pool name> -stype PureDisk -dvlist /tmp/dvlist -storage_servers <storage server name>` **Note:** You can also create the disk pool from the NetBackup web UI.
6	Update the catalog images.	Run the following command on the primary server: `/usr/openv/netbackup/bin/admincmd/bpimage -newserver "new storage server" -oldserver "old storage server" -newdiskpool "new disk pool name" -olddiskpool "old disk pool name"`
7	Delete old storage server-related configurations.	Run the following command on the old MSDP server. `/usr/openv/netbackup/bin/bp.start_all` Perform the following steps from *Recovering from an MSDP storage server failure* topic.See "Recovering from an MSDP storage server failure" on page 584. ■ Delete the storage units that use the disk pool. ■ Delete the disk pool. ■ Delete the deduplication storage server. ■ Delete the deduplication host configuration file. ■ Delete the credentials on deduplication servers.

Scenario 4: The MSDP server name has changed, the local storage is not lost, and the NetBackup catalog is not present

Step	Task	Procedure
1	Delete old storage server-related configurations.	Perform the following steps from *Recovering from an MSDP storage server failure* topic.See "Recovering from an MSDP storage server failure" on page 584. ■ Delete the storage units that use the disk pool. ■ Delete the disk pool. ■ Delete the deduplication storage server. ■ Delete the deduplication host configuration file. ■ Delete the credentials on deduplication servers.
2	Configure the new storage server.	Run the following command on the primary server: `/usr/openv/netbackup/bin/admincmd/nbdevconfig -creatests -storage_server "storage server" -stype PureDisk -media_server "media server" -st 9` **Note:** You can also create the storage server from the NetBackup web UI.
3	Reuse an existing local storage path	See "Configure/Reconfigure MSDP local storage" on page 331. **Note:** Skip these steps if the storage server was created using the NetBackup web UI.
4	Restart the storage server.	Run the following commands on the new MSDP server: `/usr/openv/netbackup/bin/bp.kill_all` `/usr/openv/netbackup/bin/bp.start_all`
5	Create disk pool for cloud LSUs.	Run the following commands on the primary server: `/usr/openv/netbackup/bin/admincmd/nbdevconfig -previewdv -storage_servers <storage server name> -stype PureDisk | grep <LSU name> > /tmp/dvlist` `/usr/openv/netbackup/bin/admincmd/nbdevconfig -createdp -dp <disk pool name> -stype PureDisk -dvlist /tmp/dvlist -storage_servers <storage server name>` **Note:** You can also create the disk pool from the NetBackup web UI.

Step	Task	Procedure
6	Prepare images for import on high-latency storage.	If the source version is 11.0 or later, MSDP automatically prepares images for import in the background. Wait for images to be ready for the import. Run the following command to check the status: `/usr/openv/pdde/pdcr/bin/tiermover --status --lsu <LSU name> [--client <client>] [--policy <policy>] [--backupid <backup ID>] [--sobins] [--active] [--lsulist] [--debug] [--verbose]` If the source version is earlier than 11.0 or the source server is down and did not finish pre-import operations, run the following command to prepare the images for import: `/usr/openv/pdde/pdcr/bin/tiermover --start --lsu <LSU name> --client <client> --policy <policy> [--backupid <backup ID>] [--retrieval Bulk|Standard|Expedited] [--verbose] [--debug]` **Note:** Import fails if the images are not ready for the import.
7	Import the images back.	Do two-phase import. See NetBackup Administrator's Guide, Volume I For the high-latency storage when the source is 11.0 or later: If import job for high-latency storage fails, it may be because the source storage did not finish preparing the images for the import before the disaster occurred. In that case, perform the step 6 again to prepare the images for import.

Scenario 5: The MSDP server name has changed, the local storage is lost, and the NetBackup catalog is present

Step	Task	Procedure
1	Configure the new storage server.	Run the following command on the primary server: `/usr/openv/netbackup/bin/admincmd/nbdevconfig -creatests -storage_server "storage server" -stype PureDisk -media_server "media server" -st 9` **Note:** You can also create the storage server from the NetBackup web UI.
2	Create an empty local LSU.	See "Configure/Reconfigure MSDP local storage" on page 331. **Note:** Skip these steps if the storage server was created using the NetBackup web UI.

Step	Task	Procedure
3	Reuse cloud LSU.	See "Reuse cloud LSU" on page 333.
4	Create disk pool for cloud LSUs.	Run the following commands on the primary server: `/usr/openv/netbackup/bin/admincmd/nbdevconfig -previewdv -storage_servers <storage server name> -stype PureDisk | grep <LSU name> > /tmp/dvlist` `/usr/openv/netbackup/bin/admincmd/nbdevconfig -createdp -dp <disk pool name> -stype PureDisk -dvlist /tmp/dvlist -storage_servers <storage server name>` **Note:** You can also create the disk pool from the NetBackup web UI.
5	Update the catalog images.	Run the following command on the primary server: `/usr/openv/netbackup/bin/admincmd/bpimage -newserver "new storage server" -oldserver "old storage server" -newdiskpool "new disk pool name" -olddiskpool "old disk pool name"`
6	Delete old storage server-related configurations.	Run the following command on the old MSDP server. `/usr/openv/netbackup/bin/bp.start_all` Perform the following steps from *Recovering from an MSDP storage server failure* topic.See "Recovering from an MSDP storage server failure" on page 584. ■ Delete the storage units that use the disk pool. ■ Delete the disk pool. ■ Delete the deduplication storage server. ■ Delete the deduplication host configuration file. ■ Delete the credentials on deduplication servers.

Scenario 6: The MSDP server name has changed, the local storage is lost, and the NetBackup catalog is not present

Step	Task	Procedure
1	Delete old storage server-related configurations.	Perform the following steps from *Recovering from an MSDP storage server failure* topic.See "Recovering from an MSDP storage server failure" on page 584. ■ Delete the storage units that use the disk pool. ■ Delete the disk pool. ■ Delete the deduplication storage server. ■ Delete the deduplication host configuration file. ■ Delete the credentials on deduplication servers.
2	Configure the new storage server.	Run the following command on the primary server: `/usr/openv/netbackup/bin/admincmd/nbdevconfig -creatests -storage_server "storage server" -stype PureDisk -media_server "media server" -st 9` **Note:** You can also create the storage server from the NetBackup web UI.
3	Create an empty local LSU.	See "Configure/Reconfigure MSDP local storage" on page 331. **Note:** Skip these steps if the storage server was created using the NetBackup web UI.
4	Reuse cloud LSU.	See "Reuse cloud LSU" on page 333.
5	Create disk pool for cloud LSUs.	Run the following commands on the primary server: `/usr/openv/netbackup/bin/admincmd/nbdevconfig -previewdv -storage_servers <storage server name> -stype PureDisk | grep <LSU name> > /tmp/dvlist` `/usr/openv/netbackup/bin/admincmd/nbdevconfig -createdp -dp <disk pool name> -stype PureDisk -dvlist /tmp/dvlist -storage_servers <storage server name>` **Note:** You can also create the disk pool from the NetBackup web UI.

Step	Task	Procedure
6	Prepare images for import on high-latency storage.	If the source version is 11.0 or later, MSDP automatically prepares images for import in the background. Wait for images to be ready for the import. Run the following command to check the status: `/usr/openv/pdde/pdcr/bin/tiermover --status --lsu <LSU name> [--client <client>] [--policy <policy>] [--backupid <backup ID>] [--sobins] [--active] [--lsulist] [--debug] [--verbose]` If the source version is earlier than 11.0 or the source server is down and did not finish pre-import operations, run the following command to prepare the images for import: `/usr/openv/pdde/pdcr/bin/tiermover --start --lsu <LSU name> --client <client> --policy <policy> [--backupid <backup ID>] [--retrieval Bulk|Standard|Expedited] [--verbose] [--debug]` **Note:** Import fails if the images are not ready for the import.
7	Import the images back.	Do two-phase import. See NetBackup Administrator's Guide, Volume I For the high-latency storage when the source is 11.0 or later: If import job for high-latency storage fails, it may be because the source storage did not finish preparing the images for the import before the disaster occurred. In that case, perform the step 6 again to prepare the images for import.

Common disaster recovery steps

Following are the common disaster recovery steps:

- See “Configure/Reconfigure MSDP local storage” on page 331.
- See “Reuse cloud LSU” on page 333.
- See “Reuse cloud LSU as read-only” on page 335.

Configure/Reconfigure MSDP local storage

Step	Task	Procedure
1	Delete the deduplication configuration.	`/usr/openv/pdde/pdconfigure/scripts/installers/PDDE_deleteConfig.sh`

Step	Task	Procedure
2	Delete the NetBackup deduplication Engine credentials on load-balancing servers.	`/usr/openv/volmgr/bin/tpconfig -delete -storage_server <sts_hostname> -stype PureDisk -sts_user_id <user_id> -all_hosts` `/usr/openv/volmgr/bin/tpconfig -add -storage_server <sts_hostname> -stype PureDisk -sts_user_id <user_id> -password <your_passwd>`
3	Redirect the command output to a file to prepare the config template.	`/usr/openv/netbackup/bin/admincmd/nbdevconfig -getconfig -storage_server <sts_hostname> -stype PureDisk >/root/local-lsu.txt` The content of the file local-lsu.txt `V7.5 "storagepath" " " string` `V7.5 "spalogin" " " string` `V7.5 "spapasswd" " " string` `....` `V7.5 "kmsservername" " " string` `V7.5 "keygroupname" " " string` `V7.5 "extendedcapabilities" " " string` `V7.5 "imagesharingincloud" "false" string`
4	Generate the config template.	Make changes to the template file `local-lsu.txt` and delete all other entries that are not needed. Parameters: `/root/local-lsu.txt` `V7.5 "storagepath" "/Storage" string` `V7.5 "spalogin" "my-user-name" string` `V7.5 "spapasswd" "my-password" string` `V7.5 "spalogretention" "90" int` `V7.5 "verboselevel" "3" int` For more information about the parameters, See “Editing an MSDP storage server configuration file” on page 197.
5	Reuse or create storage path.	`/usr/openv/netbackup/bin/admincmd/nbdevconfig -setconfig -storage_server sts_hostname -stype PureDisk -configlist /root/local-lsu.txt`

Reuse cloud LSU

Step	Task	Procedure
1	Get LSU name before you reuse cloud LSU configuration.	Run any of the following commands to get the LSUs (disk volumes) on this MSDP server. `/usr/openv/netbackup/bin/admincmd/nbdevquery -listdp -stype PureDisk -U` `/usr/openv/netbackup/bin/admincmd/nbdevquery -listdv -stype PureDisk -U` Sample output: `Disk Pool Name : my-aws-pool` `Disk Type : PureDisk` `Disk Volume Name : my-aws-lsu` Disk volume name is LSU name. In this sample, LSU name is `my-aws-lsu`.
2	Prepare a template file and save it.	Config template example 1 : `V7.5 "operation" "reuse-lsu-cloud" string` `V7.5 "cmsCredName" "your-cms-cred-name" string` `V7.5 "lsuCloudAlias" "<storageserver_lsuname>" string` `V7.5 "lsuCloudBucketName" "bucket-name" string` `V7.5 "lsuCloudBucketSubName" "lsuname" string` `V7.5 "requestCloudCacheCapacity" "1017" string` Configuration template example 2 with encryption enabled: `V7.5 "operation" "reuse-lsu-cloud" string` `V7.5 "cmsCredName" "your-cms-cred-name" string` `V7.5 "lsuCloudAlias" "<storageserver_lsuname>" string` `V7.5 "lsuCloudBucketName" "bucket-name" string` `V7.5 "lsuCloudBucketSubName" "lsuname" string` `V7.5 "lsuKmsServerName" "FQDN-KMS-server-host" string` `V7.5 "requestCloudCacheCapacity" "1017" string` If you are performing cloud LSU consolidation where the source cloud LSU resides on an older MSDP storage server than the target, add the configuration string `V7.5 "lsuMigrateCapable" "true"` to the configuration file. This step is not required if both the source and target MSDP servers are running the same version.

Step	Task	Procedure
3	Check if `lsuCloudAlias` exist.	Run the following command to list the instances to check if `lsuCloudAlias` exist. `/usr/openv/netbackup/bin/admincmd/csconfig cldinstance -i | grep <lsuname>` If an alias does not exist, run the following command to add them. `/usr/openv/netbackup/bin/admincmd/csconfig cldinstance -as -in <cloud_privder_name> -sts <storageserver> -lsu_name <lsuname> [-storage_class <storage class or tier>]` **Note:** The storage class must match the storage class that was selected during the initial configuration of LSU on the source storage server. The jobs may fail if the incorrect storage class is specified. Run the following command to find the cloud_privder_name: `/usr/openv/netbackup/bin/admincmd/csconfig cldinstance -l` You may need to include additional parameters such as storage tier, proxy configuration, authentication type, SSL settings, and son on, depending on your environment. Refer to the `csconfig cldinstance` command documentation for details on available options and their values in the *NetBackup Command Reference Guide*.
4	Reuse cloud LSU configuration.	Run the following command for each LSU to configure the cloud LSU. `/usr/openv/netbackup/bin/admincmd/nbdevconfig -setconfig -storage_server <storageserver> -stype PureDisk -configlist /root/dr-lsu.txt`
5	Recover `spad/spoold` metadata from cloud.	For each cloud LSU perform the previous four steps, and then run the following command. `/usr/openv/pdde/pdcr/bin/cacontrol --catalog clouddr <lsuname>` **Note:** The time to run this command depends on the size of the containers. Run the following command to get the catalog recovery status: `/usr/openv/pdde/pdcr/bin/cacontrol --catalog clouddrstatus <lsuname>`
6	Restart storage server.	`/usr/openv/netbackup/bin/bp.kill_all` `/usr/openv/netbackup/bin/bp.start_all`

Step	Task	Procedure
7	Start MSDP online check to recreate `refdb`.	`/usr/openv/pdde/pdcr/bin/pddecfg -a enabledataintegritycheck -d <dsid>` `/usr/openv/pdde/pdcr/bin/pddecfg -a startdatafullcheck -d <dsid>` `/usr/openv/pdde/pdcr/bin/crcontrol --processqueue --dsid <dsid>` **Note:** `-d` and `--dsid` options are optional parameters and applicable for cloud LSU only. Use `/usr/openv/pdde/pdcr/bin/pddecfg -a listcloudlsu` to get cloud LSU `dsid` value. If given `dsid` value is "0", local LSU is processed.
8	Rebuild the sampling cache.	If the P/S cache is enabled, rebuild the sampling cache. See "Rebuilding the sampling cache" on page 88.

See "Recovering the MSDP S3 IAM configurations from cloud LSU" on page 513.

Reuse cloud LSU as read-only

Step	Task	Procedure
1	Get LSU name before you reuse cloud LSU configuration.	Run any of the following commands to get the LSUs (disk volumes) on this MSDP server. `/usr/openv/netbackup/bin/admincmd/nbdevquery -listdp -stype PureDisk -U` `/usr/openv/netbackup/bin/admincmd/nbdevquery -listdv -stype PureDisk -U` Sample output: `Disk Pool Name : my-aws-pool` `Disk Type : PureDisk` `Disk Volume Name : my-aws-lsu` Disk volume name is LSU name. In this sample, LSU name is `my-aws-lsu`.

Step	Task	Procedure
2	Prepare a template file and save it.	Config template example 1 : `V7.5 "operation" "reuse-lsu-cloud" string` `V7.5 "cmsCredName" "your-cms-cred-name" string` `V7.5 "lsuCloudAlias" "<storageserver_lsuname>" string` `V7.5 "lsuCloudBucketName" "bucket-name" string` `V7.5 "lsuCloudBucketSubName" "lsuname" string` `V7.5 "requestCloudCacheCapacity" "1017" string` Configuration template example 2 with encryption enabled: `V7.5 "operation" "reuse-lsu-cloud" string` `V7.5 "cmsCredName" "your-cms-cred-name" string` `V7.5 "lsuCloudAlias" "<storageserver_lsuname>" string` `V7.5 "lsuCloudBucketName" "bucket-name" string` `V7.5 "lsuCloudBucketSubName" "lsuname" string` `V7.5 "lsuKmsServerName" "FQDN-KMS-server-host" string` `V7.5 "requestCloudCacheCapacity" "1017" string`
3	Check if `lsuCloudAlias` exist.	Run the following command to list the instances to check if `lsuCloudAlias` exist. `/usr/openv/netbackup/bin/admincmd/csconfig cldinstance -i | grep <lsuname>` If an alias does not exist, run the following command to add them. `/usr/openv/netbackup/bin/admincmd/csconfig cldinstance -as -in <cloud_privder_name> -sts <storageserver> -lsu_name <lsuname> [-storage_class <storage class or tier>]` **Note:** The storage class must match the storage class that was selected during the initial configuration of LSU on the source storage server. The jobs may fail if the incorrect storage class is specified. The jobs may fail if the incorrect storage class is specified. Run the following command to find the cloud_privder_name: `/usr/openv/netbackup/bin/admincmd/csconfig cldinstance -l`
4	Reuse cloud LSU configuration.	Run the following command for each LSU to configure the cloud LSU. `/usr/openv/netbackup/bin/admincmd/nbdevconfig -setconfig -storage_server <storageserver> -stype PureDisk -configlist /root/dr-lsu.txt`

Consolidating the cloud LSU

Starting with NetBackup version 11.1, cloud LSUs can now be consolidated from one MSDP media server to another. These steps must be repeated for each cloud LSU you plan to move.

Important considerations and recommendations before consolidation:

- Do not perform these steps on an image sharing server. Use the NetBackup web UI to add additional image sharing LSUs.
- Ensure the cloud LSU name is unique on the target MSDP media server. If the name already exists, the migration fails.
- Both the source and the target MSDP media servers should be on the latest NetBackup version.
- The following settings are not retained when a cloud LSU is moved:
  - Cloud cache capacity
  - Maximum concurrent jobs
  - Fragment size
  - Other metadata

  These must be manually reviewed and reconfigured on the target media server after the move.
- Before the consolidation, deactivate, delete, or update all backup policies that target the LSU being moved. Ensure that they now point to the cloud LSU on the target media server.
- If KMS is enabled on the source media server, it must also be enabled on the target. If moving the cloud LSU across NetBackup domains, ensure that the KMS keys are imported into the new domain.
- All policies targeting the cloud LSU must be updated or deactivated before the consolidation.

Table 7-1 Cloud LSU consolidation

Step	Task	Description
1	Deactivate backup policies.	In the NetBackup web UI, deactivate all backup policies that target the cloud LSU you plan to move. This step prevents backup or restore operations during the migration.

Table 7-1 Cloud LSU consolidation *(continued)*

Step	Task	Description
2	Fetch the original LSU information.	Use the web UI to view the disk pool details for the original cloud LSU and record the following: ■ I/O streams ■ Cloud provider ■ Region ■ Storage tier ■ Cloud cache value ■ SSL and CRL ■ Proxy settings ■ Bucket name
3	Reuse cloud LSU.	Run reuse steps on the target MSDP media server only. See "Reuse cloud LSU" on page 333. Cloud cache settings are not transferred to the new storage server. Specify `requestCloudCacheCapacity` value accordingly based on the new storage server's storage availability. In a Cloudscale deployment, all services within the media pod must be restarted to refresh the cloud LSU to storage server mapping.
4	Create a disk pool on the target server.	On the primary server, run: `/usr/openv/netbackup/bin/admincmd/nbdevconfig -previewdv -storage_servers <storage server name> -stype PureDisk | grep <LSU name> > /tmp/dvlist` /usr/openv/netbackup/bin/admincmd/nbdevconfig -createdp -dp <disk pool name> -stype PureDisk -dvlist /tmp/dvlist -storage_servers <storage server name>

Table 7-1 Cloud LSU consolidation *(continued)*

Step	Task	Description
5	Update the catalog images.	If the LSU is moved within the same NetBackup domain, run the following command on the primary server: `/usr/openv/netbackup/bin/admincmd/bpimage -newserver "target storage server" -oldserver "original storage server" -newdiskpool "new disk pool name" -olddiskpool "original disk pool name"` If moving across domains, perform a two-phase import. See *NetBackup Administrator's Guide, Volume I*.
6	Delete old storage server-related configurations.	Run the following command on the old MSDP server. `/usr/openv/netbackup/bin/bp.start_all` Perform the following steps from the *Recovering from an MSDP storage server failure* topic. See "Recovering from an MSDP storage server failure" on page 584. ■ Delete the storage units that use the disk pool. ■ Delete the disk pool. ■ The following steps should only be performed on the original MSDP media server if you plan to fully decommission it. ■ Delete the deduplication storage server. ■ Delete the deduplication host configuration file. ■ Delete the credentials on deduplication servers. See "Recovering from an MSDP storage server failure" on page 584.
7	Create a new storage unit targeting your new disk pool.	In the NetBackup web UI, create a storage unit pointing to the new disk pool. See the *NetBackup Web UI Administrator's Guide*.

Table 7-1 Cloud LSU consolidation *(continued)*

Step	Task	Description
8	Update existing policies to target the new storage unit.	In the web UI: ■ Update existing policies to use the new storage unit. Reactivate updated policies. ■ Create new policies and delete old ones.

Disaster recovery for cloud LSU in Flex Scale

When the NetBackup Flex Scale recovers from a site-based disaster, the backup data in cloud LSU can be recovered by disaster recovery of the cloud LSU.

Considerations before disaster recovery of the cloud LSU:

- The secondary NetBackup Flex Scale is ready.
  For more information, See the *Site-based disaster recovery* section of the *NetBackup Flex Scale Administrator's Guide*.
- MSDP storage server is ready and configured with the same configuration.
- KMS server is ready and key group in KMS server is ready if MSDP KMS encryption is enabled in this cloud LSU.

To perform the disaster recovery for cloud LSU

1. If the cloud instance alias does not exist, run the following command to add the alias.

   ```
   /usr/openv/netbackup/bin/admincmd/csconfig cldinstance -as -in
   <cloud_privder_name> -sts <storageserver> -lsu_name <lsuname>
   ```

2. On the NetBackup primary server, run the following command to reuse the cloud LSU. Use the same credentials, bucket name, and sub bucket that were used before the disaster recovery.

   ```
   /usr/openv/netbackup/bin/admincmd/nbdevconfig -setconfig
   -storage_server <storageserver> -stype PureDisk -configlist
   <configuration file>
   ```

   Sample configuration file:

   - If MSDP KMS encryption is enabled in this cloud LSU:

     ```
     V7.5 "operation" "reuse-lsu-cloud" string
     V7.5 "cmsCredName" "your-cms-cred-name" string
     V7.5 "lsuCloudAlias" "<storageserver_lsuname>" string
     V7.5 "lsuCloudBucketName" "bucket-name" string
     ```

```
V7.5 "lsuCloudBucketSubName" "lsuname" string
V7.5 "lsuKmsServerName" "FQDN-KMS-server-host" string
```

- If MSDP KMS encryption is disabled in this cloud LSU:

```
V7.5 "operation" "reuse-lsu-cloud" string
V7.5 "cmsCredName" "your-cms-cred-name" string
V7.5 "lsuCloudAlias" "<storageserver_lsuname>" string
V7.5 "lsuCloudBucketName" "bucket-name" string
V7.5 "lsuCloudBucketSubName" "lsuname" string
```

3. On the NetBackup primary server, get the storage server name. On the engines container with the storage server name, run the following command to get the catalog from the cloud:

   ```
   /usr/openv/pdde/pdcr/bin/cacontrol --catalog clouddr <lsuname>
   ```

   Retry this command if it fails for intermittent network issue.
4. Restart the cluster.
5. Create the disk pool for the cloud LSU.
6. Do two-phase import.

See "About the disaster recovery for cloud LSU" on page 322.

Additional steps for FortKnox Azure disaster recovery

After a disaster recovery is performed for NetBackup 10.2 or later, the refresh token of a FortKnox credential may become invalid. This can lead to jobs that use FortKnox volumes to fail. The following error message appears on the disk pool details page (Storage > Disk storage > Disk pool) of the NetBackup web UI: "Unable to retrieve associated credentials."

To fix this issue, contact support to get a new refresh token for the FortKnox credential associated with the disk volume. Then, replace the credential on the NetBackup web UI.

To replace the credential on the NetBackup web UI

1. On the left, click **Credential management.**
2. On the **Named credential** tab, click on the **Actions** menu to the right of the credential name, and click the **Edit**.
3. On the **Basic properties** page enter all the required information and click **Next**.
4. On the **Category** page, enter the storage account name and the new refresh token that was provided by the support.

5 Click **Next**.

6 On the **Review** page, review the changes and click **Finish**.

About Image Sharing using MSDP cloud

Use image sharing to share the images from your on-premises NetBackup server to another NetBackup server. The NetBackup server that is configured for image sharing is called Cloud Recovery Server (CRS). Image sharing also provides the ability to convert backed up VMs as AWS instances or Azure VHD in certain scenarios.

MSDP with image sharing is a self-describing storage server. When you configure image sharing, NetBackup stores all the data and metadata that is required to recover the images in the cloud.

Note: The Cloud Recovery Server version must be the same or later than the on-premises NetBackup version.

The following table describes the image sharing feature workflow.

Table 7-2 Image sharing workflow

Task	Description
Prepare a cloud recovery server.	You must have a virtual machine in your cloud environment and have NetBackup installed on it. You can deploy the virtual machine using one of the following ways. ■ Deploy the virtual machine using AWS Marketplace or Azure Marketplace. ■ AWS Marketplace: See Deploying NetBackup 10.0 from the AWS marketplace ■ Azure Marketplace: See Deploying NetBackup 10.0 from the Azure marketplace ■ Deploy virtual machine on-demand. ■ Create a virtual machine. ■ Install NetBackup See the *NetBackup Installation Guide*. Things to consider before you use image sharing

Table 7-2 Image sharing workflow *(continued)*

Task	Description
Configure the NetBackup KMS server.	If KMS encryption is enabled, perform the following tasks. ■ Manual KMS key transfer in Image sharing in case of NetBackup KMS ■ Manual steps in image sharing in case of external KMS
Configure image sharing on the cloud recovery server.	The NetBackup virtual machine in the cloud that is configured for image sharing is called a cloud recovery server. Perform the following steps to configure the image sharing: ■ Configure Image sharing using MSDP cloud by NetBackup web UI
Prepare images for import for high-latency storage.	This step is applicable only for high-latency storage such as Amazon Glacier, Amazon Glacier Deep Archive, and Azure Archive storage. Preparing the images for import to high-latency storage
Use the image sharing.	After you configure this NetBackup virtual machine for image sharing, you can import the images from your on-premises environment to the cloud and recover them when required. You can also convert VMs to VHD in Azure or AMI in AWS. ■ Using image sharing by NetBackup web UI ■ Using image sharing with the `nbimageshare` command ■ Things to consider before you use image sharing to convert VM image to VHD in Azure ■ Converting the VM image to VHD in Azure
Read additional information about image sharing.	Additional information about image sharing

Important features of image sharing

- Starting with NetBackup 11.1, you can add multiple cloud LSUs to an existing image sharing server.
- Multiple cloud recovery servers can be instantiated for the same MSDP cloud LSU, and each can import the catalog simultaneously.

- In a situation where MSDP cloud backed up the deduplicated data to cloud, the NetBackup catalog was available on the on-premises NetBackup server.
  Image sharing in the cloud uploads the NetBackup catalog along with the backup images and lets you restore data from the cloud without the on-premises NetBackup server.
- You can launch NetBackup on demand, which is called the cloud recovery server, and recover the backup images from the cloud.
- Image sharing discovers the backup images that are stored in cloud storage through the REST APIs, command line, or web UI, recovers the NetBackup catalog, and restores the images.
- You can use command line options or NetBackup web UI that have the function as REST APIs.
- For the imported Standard, MS Windows, and Universal share backup images, you can instantly access them with NetBackup Instant Access APIs as the exported share is in a read-only mode. For the imported VMware images, you can instantly scan them with the VMware Malware Scan APIs as the exported share is in a read-only mode.
  See “About instant access for object storage ” on page 410.
- For FortKnox, in the VM conversion procedure, a temporary bucket or blob container is created automatically. Region and the security options of the bucket are the same as the FortKnox account on the image sharing server.
  The temporary bucket or blob container name format is *vrtsonvert-<timestamp>/VRTSConvert-<timestamp>*.
- For FortKnox Amazon, MSDP-C credentials with AWS account with IAM and EC2 related permissions must be created before the VM conversion. For FortKnox Azure, MSDP-C credentials with Azure general-purpose storage accounts must be created before the VM conversion.
- For FortKnox, only the image import function uses FortKnox credentials. Before running the VM image conversion, ensure that you create MSDP-C credentials with the access credentials of your Azure/AWS account. The VM image conversion needs the access credentials of your Azure/AWS account because the Recovery Vault storage does not support creating AMI or VHD. Additionally, Azure Service Principal and AWS IAM Roles Anywhere MSDP-C credentials are not supported for VM conversion with FortKnox.

Things to consider before you use image sharing

- Before you install NetBackup, create an instance based on SUSE Linux Enterprise or RHEL 7.3 or later. You can also set up a computer based on SUSE Linux Enterprise or RHEL 7.3 or later. The recommendation is that the instance has more than 64 GB of memory, 8 CPUs.

- The HTTPS port 443 is enabled.
- Change the host name to the server's FQDN.
  In Azure virtual machine, you must change the internal host name, which is created automatically for you and you cannot get an internal host name from an IP address.
- Add the following items in the `/etc/hosts` file:
  "External IP" "Server's FQDN"
  "Internal IP" "Server's FQDN"
  For a computer, add the following items in the `/etc/hosts` file:
  “IP address” "Server's FQDN"
- (Optional) For an instance, change the search domain order in the `/etc/resolv.conf` file to search external domains before internal domains.
- For a new image sharing server, ensure that NGINX is installed and running.
  Install NGINX from Red Hat Software Collections. Refer to https://www.softwarecollections.org/en/scls/rhscl/rh-nginx114/ for instructions.
  Because the package name depends on the NGINX version, run `yum search rh-nginx` to check if a new version is available. (For NetBackup 8.3, an EEB is required if NGINX is installed from Red Hat Software Collections.)
  If you configure the storage server for image sharing before NGINX is installed and enabled, run the following command on the storage server after you install and enable NGINX:
  `/usr/openv/pdde/vpfs/bin/vpfs_config.sh --configure_byo`

Configure Image sharing using MSDP cloud by NetBackup web UI

You can access NetBackup web UI to use image sharing. For more information, refer to the *Create a Media Server Deduplication Pool (MSDP) storage server for image sharing* topic in the *NetBackup Web UI Administrator's Guide*.

Preparing the images for import to high-latency storage

If you are using image sharing for high-latency storage such as Amazon Glacier, Amazon Glacier Deep Archive, and Azure Archive storage, you must prepare the images before you import them to the cloud storage. For the source version 11.0 or later, MSDP automatically prepares images for import in the background.

To prepare the images for import to the high-latency storage

1. If the source version is earlier than 11.0 or the source server is down and did not finish pre-import operations, run the following command to prepare the images for import:

```
/usr/openv/pdde/pdcr/bin/tiermover --start --lsu <LSU name>
--client <client> --policy <policy> [--backupid <backup ID>]
[--retrieval Bulk|Standard|Expedited] [--verbose] [--debug]
```

2. If the source version is 11.0 or later, MSDP automatically prepares images for import in the background. Wait for images to be ready for the import. Run the following command to check the status:

```
/usr/openv/pdde/pdcr/bin/tiermover --status --lsu <LSU name>
[--client <client>] [--policy <policy>] [--backupid <backup ID>]
[--sobins] [--active] [--lsulist] [--debug] [--verbose]
```

Note: Import fails if the images are not ready for the import.

Using image sharing by NetBackup web UI

You can access NetBackup web UI to use image sharing. For more information, refer to the *Share images from an on-premises location to the cloud* topic in the *NetBackup Web UI Administrator's Guide*.

Using image sharing with the `nbimageshare` command

You can use the `nbimageshare` command to configure image sharing.

Run the `nbimageshare` command to list and import the virtual machine and standard images and then recover the virtual machines.

The path to access the command is: `/usr/openv/netbackup/bin/admincmd/`

For more information about the `nbimageshare` command, refer to the *NetBackup Commands Reference Guide*.

The following table lists the steps for image sharing and the command options:

Table 7-3 Steps for image sharing and the command options

Step	Command
Log on to NetBackup.	`nbimageshare --login <username> <password>` `nbimageshare --login -interact`

Table 7-3 Steps for image sharing and the command options *(continued)*

Step	Command
List all the backup images that are in the cloud.	`nbimageshare --listimage <LSU name> <MSDP image sharing server>` **Note:** In the list of images, the increment schedule type might be differential incremental or cumulative incremental.
Import the backup images to NetBackup.	Import a single image: `nbimageshare --singleimport <client> <policy> <backupID> <LSU name> <MSDP image sharing server>` Import multiple images: `--batch-import <image list file path> <LSU name> <MSDP image sharing server>` **Note:** The format of the `image_list_file_path` is same as the output of "list images". You can import the multiple images. For every 100 images, a new import job is created. You can import an already imported image. This action does not affect the NetBackup image catalog.

Table 7-3 Steps for image sharing and the command options *(continued)*

Step	Command
Recover the VM as an AWS EC2 AMI or VHD in Azure.	`nbimageshare --recovervm <LSU name> <MSDP image sharing server>` ■ Only VM images are supported. ■ This command does not support FortKnox. ■ For Azure, account should be Azure general-purpose storage accounts. ■ For AWS, the AWS account must have the following read and write permissions to S3: `"ec2:CreateTags"` `"ec2:DescribeImportImageTasks"` `"ec2:ImportImage"` `"ec2:DescribeImages"` `"iam:ListRolePolicies"` `"iam:ListRoles"` `"iam:GetRole"` `"iam:GetRolePolicy"` `"iam:CreateRole"` `"iam:PutRolePolicy"`

Manual KMS key transfer in Image sharing in case of NetBackup KMS

When KMS encryption is enabled, you can share the images in the cloud storage to the cloud recovery server with manual KMS key transfer.

On-premises side:

1. Storage server: Find the key group name for the given Storage server.

   Find `contentrouter.cfg` in `/etc/pdregistry.cfg`

   Find the key group name is in `contentrouter.cfg` under `[KMSOptions]`

   (Example `KMSKeyGroupName=amazon.com:test1`)

2. NetBackup primary server: Exports the key group with a passphrase to a file:

   ```
   /usr/openv/netbackup/bin/admincmd/nbkmsutil -export -key_groups
   <key-group-name> -path <key file path>
   ```

cloud recovery server (cloud side):

1. Copy the exported key to the cloud recovery server.
2. Configure the KMS server.

```
/usr/openv/netbackup/bin/nbkms -createemptydb
/usr/openv/netbackup/bin/nbkms
/usr/openv/netbackup/bin/nbkmscmd -discovernbkms -autodiscover
```

3. Import keys to KMS service.

   ```
   /usr/openv/netbackup/bin/admincmd/nbkmsutil -import -path <key
   file path> -preserve_kgname
   ```

4. Configure the cloud recovery server using NetBackup web UI.

On-premises KMS key changes:

In the case of KMS key changes for the given group for on-premises storage server after the cloud recovery server is set up, you must export the key file from the on-premises KMS server and import that key file on the cloud recovery server.

1. On-premises NetBackup primary server:

   Exports the key group with a passphrase to a file:

   ```
   /usr/openv/netbackup/bin/admincmd/nbkmsutil -export -key_groups
   <key-group-name> -path <key file path>
   ```

2. Cloud recovery server:

   ```
   /usr/openv/netbackup/bin/admincmd/nbkmsutil -deletekg -kgname
   <key-group-name> -force
   ```

   ```
   /usr/openv/netbackup/bin/admincmd/nbkmsutil -import -path <key
   file path> -preserve_kgname
   ```

Manual steps in image sharing in case of external KMS

If an on-premises storage server is configured to use keys from an external KMS server, then make sure that the same KMS server is configured on the cloud recovery server before configuring image sharing through the web UI. To know more about configuring an external KMS server in NetBackup, refer to *NetBackup Security and Encryption Guide*.

Make sure that the external KMS server is reachable from the cloud recovery server on a specific port.

Additional information about image sharing

- It is recommended that you launch a cloud recovery server on demand and do not upgrade it.

- Do not use `nbdevconfig` to modify cloud LSU or add new cloud LSU in the image sharing server as it might cause an issue in the image sharing server (cloud recovery server). Use the NetBackup web UI to add new cloud LSUs or modify existing ones for image sharing.
  If KMS encryption is enabled in the on-premises side after image sharing server is configured, the encrypted image cannot be imported by this image sharing server.
- For each cloud LSU configured with KMS that is added to the image sharing server, you must import the corresponding KMS keys into the image sharing server.
- Cloud LSU requires free disk space. When you configure image sharing server, ensure that you have sufficient disk space in the default mount point or storage.
- After the image is imported in the image sharing server, the image catalog exists in the image sharing server. If the image is expired on the on-premises NetBackup domain, then restoring the image to the image sharing server fails even though the image catalog exists in the image sharing server.
- The imported image expiration time is the time for which the imported image catalog exists in the image sharing server. If the image expires in the image sharing server, the image catalog in the image sharing server is removed but the image data in the cloud storage is not removed.
- You can restore any image that you import in the image sharing server. Only VM images in AWS and Azure can be recovered because they can be converted into EC2 instances in AWS or VHD in Azure. VM images in other cloud storages cannot be converted, and can only be restored. You can recover only the VM images that are full backup images or accelerator-enabled incremental backup images.
- Image sharing supports many policy types.
  See the NetBackup compatibility list for the latest information on the supported policy types.
- After the image sharing is configured, the storage server is in a read-only mode. Some MSDP commands are not supported.
- For information on the VM recovery limitations in AWS, refer to the AWS VM import information in AWS help.
- You can configure the maximum active jobs when the images are imported to cloud storage.
  Modify the file path `/usr/openv/var/global/wsl/config/web.conf` to add the configuration item as `imageshare.maxActiveJobLimit`.
  For example, `imageshare.maxActiveJobLimit=16`.
  The default value is 16 and the configurable range is 1 to 100.

If the import request is made and the active job count exceeds the configured limit, the following message is displayed:
"Current active job count exceeded active job count limitation".

- The images in cloud storage can be shared. If Amazon Glacier, Deep Archive or Azure Archive is enabled, you cannot use image sharing for the storage servers running versions earlier than 11.0.
- If Amazon Glacier, Deep Archive or Azure Archive is enabled and the LSU was previously migrated from Cloud Catalyst, you cannot use image sharing.
- Regarding the errors about role policy size limitation in AWS:
  Errors that occur when the role policy size exceeds the maximum size is an AWS limitation. You can find the following error in a failed restore job:

  ```
  "error occurred (LimitExceeded) when calling the PutRolePolicy operation:
  Maximum policy size of 10240 bytes exceeded for role vmimport"
  ```

  Workaround:

  - You can change the maximum policy size limit for the `vmimport` role.
  - You can list and delete the existing policies using the following commands:

    ```
    aws iam list-role-policies --role-name vmimport
    aws iam delete-role-policy --role-name vmimport --policy-name
    <bucketname> -vmimport
    ```

- The recover operation with AWS provider includes the AWS import process. Therefore, a vmdk image cannot be recovered concurrently in two restore jobs at the same time.
- In AWS, the image sharing feature can recover the virtual machines that satisfy the Amazon Web Services VM import prerequisites.
  For more information about the prerequisites, refer to the following article:
  https://docs.aws.amazon.com/vm-import/latest/userguide/vmie_prereqs.html
- If you cannot obtain the administrator password to use an AWS EC2 instance that has a Windows OS, the following error is displayed:

  ```
  Password is not available. This instance was launched from a custom
  AMI, or the default password has changed. A password cannot be
  retrieved for this instance. If you have forgotten your password,
  you can reset it using the Amazon EC2 configuration service. For
  more information, see Passwords for a Windows Server Instance.
  ```

  This error occurs after the instance is launched from an AMI that is converted using image sharing.

  For more information, refer to the following articles:

  - Amazon Elastic Compute Cloud Common Messages
  - How to migrate your on-premises domain to AWS Managed Microsoft AD using ADMT
- You cannot cancel an import job on the cloud recovery server.
- If there is data optimization done on the on-premises image, you might not be able to restore the image that you have imported on the cloud recovery server. You can expire this image, import it again on the image-sharing server, and then restore the image.
- After the backup job, duplication job, or AIR import job completes, you can import the images on a cloud recovery server. The images that are created by User-Archive job cannot be imported.
- AIR is not supported for the image sharing server.
- If you want to convert a VM image again, you must delete the VHD from the Azure blob.

Things to consider before you use image sharing to convert VM image to VHD in Azure

Image sharing with Azure provider support converting VMware virtual machine to Azure VHD, which is uploaded to Azure storage blob. You can use Azure web portal to create a VM based on VHD. Image sharing does not add additional limitations about VM conversion, but Azure has the following prerequisites on source VMs:

- Source virtual machine OS Type
  The following guest operating systems in source virtual machine are supported:
  - Windows 10 Series
  - Windows 2012 R2 Series
  - Windows 2016 Series
  - Windows 2019 Series
  - Windows 2022 Series
  - RHEL 7.6, 7.7, 7.9, 8.6
  - Ubuntu 18.04
  - SUSE 12SP4, 15SP4

  For other operation systems, see Supported platforms.
  For non-endorsed distributions, verify that the source VM meets the requirements for non-endorsed distributions before you convert a VM. This verification is important because Linux VMs that are based on an endorsed distribution of

Microsoft Azure have the prerequisites that enable them to run on Azure, but the VMs that originate from other hypervisors might not. For more information, see Information for Non-Endorsed Distributions.

- Hyper-V Drivers in source virtual machine
  For Linux, the following Hyper-V drivers are required on the source VM:
  - hv_netvsc.ko
  - hv_storvsc.ko
  - hv_vmbus.ko

  You may need to rebuild the initrd so that required kernel modules are available on the initial ramdisk. The mechanism for rebuilding the initrd or initramfs image may vary depending on the distribution. Many distributions have these built-in drivers available already. For Red Hat or CentOS, the latest Hyper-V drivers (LIS) may be required if the built-in drivers do not work well. For more information, see Linux Kernel requirements.

  For example, before you perform a backup for a Linux source VM that runs CentOS or Red Hat, verify that required Hyper-V drivers are installed on the source VM. Those drivers must be present on the source VM backup to start the VM after conversion.
  - Take a snapshot of the source VM..
  - Run the following command to modify the boot image:
    ```
    sudo dracut -f -v -N
    ```
  - Run the following command to verify that Hyper-V drivers are present in the boot image:
    ```
    lsinitrd | grep hv
    ```
  - Verify that no dracut conf files (for example, `/usr/lib/dracut/dracut.conf.d/01-dist.conf`) contain the following line:
    ```
    hostonly="yes"
    ```
  - Run a new backup to use for the conversion.
- Disk
  - The OS in source VMs is installed on the first disk of the source VMs. Do not configure a swap partition on the operating system disk. See Information for Non-endorsed Distributions
  - Multiple Data disks attached to new VM created by converted VHD will be in offline status for Windows and unmounted for Linux. Need to make them online and mount manually after conversion.

  - After creating a VM by converted VHD, one extra temporary storage disk whose size is determined by the VM size may be added by Azure in both Linux and Windows systems. For more information, see Azure VM Temporary Disk.

- Networking
  If the source VM has multiple network interfaces, only one interface will be kept available in the new VM created by converted VHD.
  Linux: The name of the primary network interface on source VMs must be eth0 for endorsed Linux distributions. If not, it is unable to connect new VM created by converted VHD, and some manual steps need to be done on the converted VHDs. For more information, see Can't connect to Azure Linux VM through network.
  Windows: Enable Remote Desktop Protocol (RDP) on the source VM. Some windows systems need to disable the firewall in source VMs, otherwise unable to connect remotely.

- Azure account
  When you convert VMDK to VHD, Azure account in image sharing using MSDP cloud should be Azure general-purpose storage accounts. See Storage account overview.

Note: For VM conversion, if the image sharing volume is FortKnox, only access credentials are supported, Azure Service Principal or AWS IAM anywhere credentials are not supported.

Converting the VM image to VHD in Azure

You can covert the following VM images to VHD in Azure using the image sharing:

Table 7-4 VM image to VHD conversion

VM image operating system	Description
Windows VM	See “Converting the Windows VM image to VHD ” on page 355.
RHEL7.6 VM	See “Converting the RHEL7.6 VM image to VHD ” on page 355.
SUSE 12 SP4 VM	See “Converting SUSE 12 SP4 VM image to VHD” on page 357.
RHEL 8.6 VM	See “Converting RHEL 8.6 VM image to VHD ” on page 358.

Table 7-4 VM image to VHD conversion *(continued)*

VM image operating system	Description
SLES 15 SP4 VM	See "Converting SLES 15 SP4 VM image to VHD " on page 360.

Converting the Windows VM image to VHD

To convert the Windows VM image to VHD

1. Enure that you enable Remote Desktop Connection on your source VM before backup.
2. Perform a new full backup of the source VM,
3. Prepare an image sharing server and configure image sharing feature with azure account.
4. Import the backup image and perform the conversion.
5. Verify the converted vhd files.

   In Azure web portal:

   - Create a disk with the converted .vhd file.
   - Create a VM with the previous disk.
     Navigate to **Disks > Created disk > Create VM**. With default Networking & Disks & Management settings, enable boot diagnostics.
   - Log in to the converted VM through RDP.

Converting the RHEL7.6 VM image to VHD

Pre-requisites:

- Source VM OS volume must use MBR partitioning rather than GPT.
- Use the persistent naming (file system label or UUID) in `fstab` configuration. Most distributions provide the `fstab nofail` or `nobootwait` parameters. These parameters enable a system to boot when the disk fails to mount at startup.
- Ensure that the operating system is installed on the first disk of the source VM. Do not configure a swap partition on the operating system disk. See Information for Non-endorsed Distributions.
- We recommend that the network interface in the source VM uses DHCP and enabled on boot. See Add, change, or remove IP addresses for an Azure network interface.
- See Prepare a Red Hat-based virtual machine for Azure.

To convert the RHEL7.6 VM image to VHD

1. Install latest LIS 4.3.5.

```
tar -xzf lis-rpms-4.3.5.x86_64.tar.gz

cd LISISO

./install

reboot
```

2. Rebuild `initramfs` image file.

```
cd /boot

cp initramfs-`uname -r`.img initramfs-`uname -r`.img.bak
```

Run the following command to open the `dracut.conf` file:

```
vi /etc/dracut.conf
```

Uncomment the line `#add_drivers+=""`

Add the following drivers to the line, separating each module with the space.

```
hv_netvsc hv_storvsc hv_vmbus
```

Example,

```
# additional kernel modules to the default.
add_drivers+="hv_netvsc hv_storvsc hv_vmbus"
```

Create new initial ramdisk images with new modules.

```
dracut -f -v -N
```

Run any of the following commands to check if the new modules exist in new initial ramdisk images.

```
lsinitrd | grep -i hv

lsinitrd -f /boot/initramfs-`uname -r`.img | grep -i hv

modinfo hv_netvsc hv_storvsc hv_vmbus
```

3 Rename the network interface to **eth0** and enable it on boot.

In the network interface configuration file, configure: `ONBOOT=yes`.

For example,

```
mv /etc/sysconfig/network-scripts/ifcfg-ens192
/etc/sysconfig/network-scripts/ifcfg-eth0
```

```
sed -i 's/ens192/eth0/g' /etc/sysconfig/network-scripts/ifcfg-eth0
```

In the file `/etc/default/grub`, change the line `GRUB_CMDLINE_LINUX="xxxxxxx"` to `GRUB_CMDLINE_LINUX="xxxxxxx net.ifnames=0 biosdevname=0"`

```
grub2-mkconfig -o /boot/grub2/grub.cfg
```

4 Perform a new full backup of the source VM,

5 Prepare the image sharing server and configure the image sharing feature with Azure account.

6 Import the backup image and perform the conversion.

7 Verify the converted vhd files.

In the Azure web portal:

- Create a disk with the converted .vhd file.
- Create a VM with the previous disk.
  Navigate to **Disks > Created disk > Create VM**. With default Networking & Disks & Management settings, enable boot diagnostics.
- Log in to the converted VM through RDP.

Converting SUSE 12 SP4 VM image to VHD

Pre-requisites:

- Source VM OS volume must use MBR partitioning rather than GPT.
- Use the persistent naming (file system label or UUID) in `fstab` configuration. Most distributions provide the `fstab nofail` or `nobootwait` parameters. These parameters enable a system to start when the disk fails to mount at startup.
- Ensure that the operating system is installed on the first disk of the source VM. Do not configure a swap partition on the operating system disk. See Information for Non-endorsed Distributions.
- We recommend that the network interface in the source VM uses DHCP and enabled on boot. See Add, change, or remove IP addresses for an Azure network interface.

To convert the SUSE 12 SP4 VM image to VHD

1. Ensure that the required modules are installed.

   - `lsinitrd -f /boot/initramfs-`uname -r`.img | grep -i hv`

     Or

     ```
     modinfo hv_vmbus hv_storvsc hv_netvsc
     reboot
     ```

   - Rebuild `initrd`.

     ```
     cd /boot/
     cp initrd-$(uname -r) initrd-$(uname -r).backup
     mkinitrd -v -m "hv_vmbus hv_netvsc hv_storvsc" -f
     /boot/initrd-$(uname -r) $(uname -r)
     ```

2. Check the network interface name eth0 and enabled on boot.

   /etc/sysconfig/network/ifcfg-eth0 contains the record:

   ```
   STARTMODE='auto'
   ```

3. Perform a new full backup of the source VM,
4. Prepare an image sharing server and configure the image sharing feature with azure account.
5. Import the backup image and perform the conversion.
6. Verify the converted vhd files.

   In the Azure web portal:

   - Create a disk with the converted .vhd file.
   - Create a VM with the previous disk.
     Navigate to **Disks > Created disk > Create VM**. With default **Networking & Disks & Management** settings, enable boot diagnostics.
   - Log in to the converted VM through RDP.

Converting RHEL 8.6 VM image to VHD

Pre-requisites:

- The boot options of the source VMs are BIOS or UEFI. Use the standard partitions rather than a logical volume manager (LVM), which is the default for many installations.
- Use the persistent naming (file system label or UUID) in `fstab` configuration.
- Ensure that the operating system is installed on the first disk of the source VM. Do not configure a swap partition on the operating system disk.

- We recommend that the network interface in the source VM uses DHCP and enabled on start.

See Prepare a Red Hat-based virtual machine for Azure

To convert the RHEL 8.6 VM image to VHD

1 Install Hyper-V device drivers and rebuild the `initramfs` image file.

Check if the Hyper-V drivers (hv_netvsc, hv_storvsc, hv_vmbus) are installed or not.

```
lsinitrd | grep hv
```

If they are not installed, perform the following steps.

- Back up the previous `initramfs` image file.

```
cd /boot
cp initramfs-`uname -r`.img initramfs-`uname -r`.img.bak
```

- Create a file `hv.conf` under the directory `/etc/dracut.conf.d`. Add the following driver parameters to the `hv.conf` file.

```
add_drivers+=" hv_vmbus "
add_drivers+=" hv_netvsc "
add_drivers+=" hv_storvsc "
add_drivers+=" nvme "
```

Note: Add the spaces between the quotes and the driver name.

- Create new initial ramdisk images with new modules.

```
dracut -f -v -N -regenerate-all
```

Check if the new modules exist in new initial ramdisk images.

```
lsinitrd | grep -i hv
```

2 Rename the network interface to the name **eth0** and enable the NIC on boot.

Azure Linux VMs use traditional NIC names by default.

In the network interface configuration file, configure `ONBOOT=yes`.

For example,

```
mv /etc/sysconfig/network-scripts/ifcfg-ens192
/etc/sysconfig/network-scripts/ifcfg-eth0 sed -i 's/ens192/eth0/g'
/etc/sysconfig/network-scripts/ifcfg-eth0
```

3 Regenerate the `grub.cfg` for the kernel boot options.

- To use the traditional NIC names in the file `/etc/default/grub`, change the line `GRUB_CMDLINE_LINUX="xxxxxxx"` to `GRUB_CMDLINE_LINUX="xxxxxxx net.ifnames=0"`
  Remove the following parameters if they exist: `rhgb quiet crashkernel=auto`
- Regenerate the `grub.cfg` file.
  On a BIOS-based computer: `grub2-mkconfig -o /boot/grub2/grub.cfg`
  On a UEFI-based computer: `grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg`

4. Perform a new full backup of the source VM.
5. Prepare the image sharing server and configure the image sharing feature with azure account.
6. Import the backup image and perform the conversion.
7. Verify the converted VHD files.

   In the Azure web portal:

   - Create a disk with the converted .vhd file.
   - Create a VM with the previous disk.
     Navigate to **Disks > Created disk > Create VM**. With default **Networking & Disks & Management** settings, enable the boot diagnostics.
   - Log in to the converted VM through SSH.

Converting SLES 15 SP4 VM image to VHD

Pre-requisites:

- The boot options of the source VMs are BIOS or UEFI. Use the standard partitions rather than a logical volume manager (LVM), which is the default for many installations.
- Use the persistent naming (file system label or UUID) in `fstab` configuration.
- Ensure that the operating system is installed on the first disk of the source VM. Do not configure a swap partition on the operating system disk.
- We recommend that the network interface in the source VM uses DHCP and enabled on start.

To convert the RHEL 8.6 VM image to VHD

1. Install Hyper-V device drivers and rebuild the `initramfs` image file.

Check if the Hyper-V drivers (hv_netvsc, hv_storvsc, hv_vmbus) are installed or not.

```
lsinitrd | grep hv
```

If they are not installed, perform the following steps.

- Back up the previous `initramfs` image file.

  ```
  cd /boot
  cp initramfs-`uname -r`.img initramfs-`uname -r`.img.bak
  ```

- Create a file `hv.conf` under the directory `/etc/dracut.conf.d`. Add the following driver parameters to the `hv.conf` file.

  ```
  add_drivers+=" hv_vmbus "
  add_drivers+=" hv_netvsc "
  add_drivers+=" hv_storvsc "
  add_drivers+=" nvme "
  ```

  Note: Add the spaces between the quotes and the driver name.

- Create new initial ramdisk images with new modules.
  `dracut -f -v -N -regenerate-all`
  Check if the new modules exist in new initial ramdisk images.
  `lsinitrd | grep -i hv`

2 Check that the network interface name is **eth0**. Ensure that the network interface is using DHCP, and it is enabled on boot.

`/etc/sysconfig/network/ifcfg-eth0` contains the following:

```
BOOTPROTO='dhcp'
STARTMODE='auto'
```

3 Regenerate the `grub.cfg` to ensure that console logs are sent to the serial port.

- To use the traditional NIC names in the file `/etc/default/grub`, change the line `GRUB_CMDLINE_LINUX="xxxxxxx"` to `GRUB_CMDLINE_LINUX="xxxxxxx net.ifnames=0"`
  Remove the following parameters if they exist: `rhgb quiet crashkernel=auto`
- Regenerate the `grub.cfg` file.
  On a BIOS-based computer: `grub2-mkconfig -o /boot/grub2/grub.cfg`

On a UEFI-based computer: `grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg`

4. Perform a new full backup of the source VM.
5. Prepare the image sharing server and configure the image sharing feature with azure account.
6. Import the backup image and perform the conversion.
7. Verify the converted VHD files.

   In the Azure web portal:

   - Create a disk with the converted .vhd file.
   - Create a VM with the previous disk.
     Navigate to **Disks > Created disk > Create VM**. With default **Networking & Disks & Management** settings, enable boot diagnostics.
   - Log in to the converted VM through SSH.

About restore from a backup in Microsoft Azure Archive

After initiating restore, the rehydrate process in Microsoft Azure Archive takes time. For more information refer to the Microsoft Azure documentation. The rehydrate process completes when the data is transitioned to the Hot tier. The number of days specified while the configuring LSU measures the time the data will stay on the Hot tier. After this the data is transitioned to the Archive tier.

The number of days you keep the data in the Hot tier impacts the cloud provider cost.

You can modify the value of the rehydration period using the `csconfig CLI. -post_rehydration_period`command.

About FortKnox for NetBackup

FortKnox for NetBackup provides a cloud-based storage-as-a-service (SaaS) offering that provides a seamless, secondary storage option. This feature offers a single, flexible repository for on-premises to your public cloud workloads. FortKnox is available in the web UI so you can simplify the provisioning, management, and monitoring of cloud storage resources and retention policies.

An accurate representation of your Recovery Vault storage is displayed on the FortKnox storage usage overview page in Cohesity Alta View. You can view the

storage details and usage details for your configured cloud service providers. The storage details are normally 24-48 hours delayed due to the periodic data refresh cycle that is provided by the cloud service providers.

For more information about FortKnox for NetBackup, refer to the following links:

Veritas Alta Recovery Vault Deployment Guide

Explore Recovery Vault

Configuring FortKnox for NetBackup Azure and Azure Government

Use the following procedure to configure FortKnox for NetBackup for Azure and Azure Government.

Note: FortKnox supports multiple options. For FortKnox Azure and Azure Government options in the web UI, you must contact your Cohesity NetBackup account manager for credentials or with any questions.

Table 7-5 Steps for configuring FortKnox for Azure and Azure Government

Steps	Task	Instructions
Step 1	Retrieve credentials.	Retrieve the FortKnox credentials from your Cohesity NetBackup account manager.
Step 2	(Optional) Create an MSDP storage server if one does not exist.	See “Configuring media server deduplication in NetBackup” on page 63.

Table 7-5 Steps for configuring FortKnox for Azure and Azure Government *(continued)*

Steps	Task	Instructions
Step 3	Add a disk pool.	In the NetBackup web UI, create a disk pool. Follow the procedure in *Create a disk pool* in the *NetBackup Web UI Administrator's Guide*. ■ In the **Volumes** step: ■ Select the option of **FortKnox Azure** or **FortKnox Azure Government** from the **Cloud storage provider** drop-down. ■ Select the appropriate region. ■ In the **Associate credentials** section, select **Add a New Credential** and enter the **Storage account** and a **Refresh token** provided by the provisioning team. Or, you can use **Select existing credentials** if credentials exist for Azure. The **Credential name** must use alphanumeric characters with hyphens or underscores and cannot contain spaces or illegal characters. FortKnox Azure provides the following two storage classes: ■ FortKnox Azure Warm Vault ■ FortKnox Azure Cold Vault Select the appropriate storage tier depending on the storage deployment. The actual storage tier is determined by the NetBackup during the configuration. **Note:** You may see a container named `fortknox-azure-29cb2539-39ce-427e-8e72-48bf62177bc3` that is created for NetBackup use. Ensure that you do not create a bucket with the same name. If you create a bucket with the same name, the disk pool configuration fails.
Step 4	Add a storage unit.	In the NetBackup web UI, create a storage unit. Follow the procedure in *Create a storage unit* in the *NetBackup Web UI Administrator's Guide*. When you create the storage unit, select the **Media Server Deduplication Pool (MSDP)** option. In the **Disk pool** step, select the disk pool that was created in Step 3.

Note: If an update to the refresh token for an existing storage account is needed, you must edit the credentials that are associated with the storage account. Use the web UI and update the refresh token within **Credential management**.

You cannot have multiple credentials for the same storage account. Credentials must be unique to the storage account. If you do not have unique credentials, you can encounter issues such as the disk volume going down or backup and restore failures to that disk volume.

Note: Use the original storage account to update the FortKnox CMS credentials.

Configuring FortKnox for NetBackup Azure and Azure Government using the CLI

Use the following procedure to configure FortKnox for NetBackup for Azure and Azure Government using the CLI.

Table 7-6 Steps for configuring FortKnox for Azure and Azure Government with the CLI

Steps	Task	Instructions
Step 1	Retrieve credentials.	Retrieve FortKnox credentials from your Cohesity NetBackup account manager.

Table 7-6 Steps for configuring FortKnox for Azure and Azure Government with the CLI *(continued)*

Steps	Task	Instructions
Step 2	Add credentials using the **Credential management** option.	Log into NetBackup web UI and perform the following: **1** On the left, click **Credential management**. **2** On the **Named credentials** tab, click **Add** and provide the following properties: ■ **Credential name** The **Credential name** must use alphanumeric characters with hyphens or underscores and cannot contain spaces or illegal characters. ■ **Tag** ■ **Description** **3** Click **Next**. **4** In the drop-down, select **FortKnox**. **5** Click **FortKnox Azure**. **6** Add the **Storage account** and **Refresh token**. **7** Select or add a role that can access this credential. **8** Review the information and click **Finish**.
Step 3	Create an MSDP storage server.	See "Configuring media server deduplication in NetBackup" on page 63.

Table 7-6 Steps for configuring FortKnox for Azure and Azure Government with the CLI *(continued)*

Steps	Task	Instructions
Step 4	Create a cloud instance alias.	Use the following examples depending on your environment: ■ Creating an FortKnox Azure cloud instance alias: `/usr/openv/netbackup/bin/admincmd/csconfig cldinstance -as -in FortKnox-Azure -sts` **<storage server>** `-lsu_name` **<lsu name>** ■ Creating an FortKnox Azure Cold Vault cloud instance alias: `/usr/openv/netbackup/bin/admincmd/csconfig cldinstance -as -in FortKnox-Azure -sts` **<storage server>** `-lsu_name` **<lsu name>** `-storage_tier FortKnox-Azure-Cold-Vault -post_rehydration_period 3` The cloud alias name is **<storage server>_<lsu name>**, and is used to create a bucket.
Step 5	(Optional) Create a new bucket.	Create a new bucket if needed. `/usr/openv/netbackup/bin/nbcldutil -createbucket -storage_server` **<storage server>_<lsu name>** `-username` **<cloud user>** `-bucket_name` **<bucket name>**

Table 7-6 Steps for configuring FortKnox for Azure and Azure Government with the CLI *(continued)*

Steps	Task	Instructions
Step 6	Create a configuration file, then run `nbdevconfig` command.	

Table 7-6 Steps for configuring FortKnox for Azure and Azure Government with the CLI *(continued)*

Steps	Task	Instructions
		Configuration file content for adding a new cloud LSU (configuration setting and description): ■ V7.5 `"operation" "add-lsu-cloud"` string - Specifies the value `"add-lsu-cloud"` for adding a new cloud LSU. ■ V7.5 `"lsuName" " "` string - Specifies the LSU name. ■ V7.5 `"cmsCredName" " "` string - Specifies the **Credential name** created using credential management. ■ V7.5 `"lsuCloudBucketName" " "` string - Specifies the cloud bucket name. ■ V7.5 `"lsuCloudBucketSubName" " "` string - Multiple cloud LSUs can use the same cloud bucket. This value distinguishes different cloud LSUs. ■ V7.5 `"lsuEncryption" " "` string - Optional value and the default is `NO`. Sets the encryption property for the current LSU. ■ V7.5 `"lsuKmsEnable" " "` string - Optional value and the default is `NO`. Enables KMS for the current LSU. ■ V7.5 `"lsuKmsKeyGroupName" " "` string - Optional value. Key group name is shared among all LSUs. Key group name must have valid characters: `A-Z, a-z, 0-9, _ (underscore), - (hyphen), : (colon), . (period), and space`. ■ V7.5 `"lsuKmsServerName" " "` string - Optional value. KMS server name is shared among all LSUs. ■ V7.5 `"lsuKmsServerType" " "` string - Optional value. Example of a configuration file with encryption disabled: `V7.5 "operation" "add-lsu-cloud" string` `V7.5 "lsuName" "nbrvltazure1" string` `V7.5 "cmsCredName" "RVLT-creds" string` `V7.5 "lsuCloudBucketName" "bucket1" string` `V7.5 "lsuCloudBucketSubName" "sub1" string` Example of a configuration file with encryption enabled: `V7.5 "operation" "add-lsu-cloud" string` `V7.5 "lsuName" "nbrvltazure2" string` `V7.5 "cmsCredName" "RVLT-creds" string` `V7.5 "lsuCloudBucketName" "bucket1" string`

Table 7-6 Steps for configuring FortKnox for Azure and Azure Government with the CLI *(continued)*

Steps	Task	Instructions
		`V7.5 "lsuCloudBucketSubName" "sub2" string` `V7.5 "lsuEncryption" "YES" string` `V7.5 "lsuKmsEnable" "YES" string` `V7.5 "lsuKmsKeyGroupName" "test" string` `V7.5 "lsuKmsServerName" "test" string` **Note:** All encrypted LSUs in one storage server must use the same `keygroupname` and `kmsservername`. If you use the `nbdevconfig` command to add a new encrypted cloud LSU and one exists in this MSDP, the `keygroupname` must be the same as the `keygroupname` in the previous encrypted LSU. After you create the configuration file, run the `nbdevconfig` command: `/usr/openv/netbackup/bin/admincmd/nbdevconfig -setconfig -storage_server` **`<storage server>`** `-stype PureDisk -configlist` **`<configuration file path>`** **Note:** The parameter **<storage server>** must be the same as the parameter **<storage server>** in Step 4.

Table 7-6 Steps for configuring FortKnox for Azure and Azure Government with the CLI *(continued)*

Steps	Task	Instructions
Step 7	Create disk pool.	Create disk pool by running the `nbdevconfig` command. The following are examples of using the `nbdevconfig` command: Example 1: `/usr/openv/netbackup/bin/admincmd/nbdevconfig -previewdv -storage_servers` **`<storage server name>`** `-stype PureDisk	grep` **`<LSU name>`** `> /tmp/dvlist` Example 2: `/usr/openv/netbackup/bin/admincmd/nbdevconfig -createdp -dp` **`<disk pool name>`** `-stype PureDisk -dvlist /tmp/dvlist -storage_servers` **`<storage server name>`** **Note:** You can also create the disk pool from the NetBackup web UI.
Step 8	Create storage unit.	Create storage unit by using `bpstuadd` command. The following are examples of using the `bpstuadd` command: `/usr/openv/netbackup/bin/admincmd/bpstuadd -label` **`<storage unit name>`** `-odo 0 -dt 6 -dp` **`<disk pool name>`** `-nodevhost` **Note:** You can also create the storage server from the NetBackup web UI.

Note: If an update to the refresh token for an existing storage account is needed, you must edit the credentials that are associated with the storage account. Use the web UI and update the refresh token within **Credential management**.

You cannot have multiple credentials for the same storage account. Credentials must be unique to the storage account. If you do not have unique credentials, you can encounter issues such as the disk volume going down or backup and restore failures to that disk volume.

About `csconfig cldinstance` changes for FortKnox for Azure and Azure Government

The `csconfig cldinstance` command displays the `Need Token Renew` flag that retrieves alias information (Yes/No). When `Yes`, FortKnox expects the storage account and refresh token credentials instead of storage account and access key.

The cloud instance has an option to disable (0) or enable (1) the need token renew (-ntr) option which has a default value of yes (1) for `FortKnox-Azure` and `FortKnox-Azure-Gov`.

Example usage of `csconfig cldinstance` with `-ntr`:

```
csconfig cldinstance -us -in <instance name> -sts <alias name> -ntr <0,1>
```

Note: When you add the cloud LSU on a back-level media server using the CLI, the `-ntr` option must be set to `No` (0). You must set the option to No because older versions of the media server don't have support for token based credentials. When you use a NetBackup storage server version 10.2 or newer, the cloud alias instance must have the `-ntr` option set to `Yes`. The setting cannot be set to `No`.

About `nbcldutil` changes for FortKnox for Azure and Azure Government

The `nbcldutil` command has new inputs for the `-createbucket` and `-validatecreds` options when you configure FortKnox for Azure and Azure Government.

Example usage:

```
nbcldutil -createbucket storage_server storage-server-name_lsu-name
-username rvlt-creds -bucket_name sl-bucket-cli
```

Instead of putting the storage account name for `-username`, use the name of the credentials created using Credential Management. Also, when prompted for a password, provide a dummy input because no password is needed.

About `msdpcldutil` changes for FortKnox for Azure and Azure Government

To use this utility, you must create a credential name using the NetBackup web UI and also create a cloud alias using the `csconfig` command similar to the following example:

```
/usr/openv/netbackup/bin/admincmd/csconfig cldinstance
-as -in FortKnox-Azure -sts <storage_server_name>
```

```
-stype PureDisk -lsu_name test1

Successfully added storage server(s): <storage_server_name>_test1
```

Added the `--enable_sas` option to use for FortKnox Azure. Additionally, if the `--enable _sas` option is used you must export the following environment variables:

- `MSDPC_MASTER_SERVER` - This option is name of the NetBackup primary server.
- `MSDPC_ALIAS` - This option is the cloud alias created using `csconfig`.
- `MSDPC_ACCESS_KEY` - A credential name and `MSDPC_SECRET_KEY` is a dummy string.

Example output:

```
export MSDPC_PROVIDER=vazure
export MSDPC_REGION="East US"
export MSDPC_ENDPOINT="https://<storage-account>.blob.core.windows.net/"
export MSDPC_ACCESS_KEY=<credential name>
export MSDPC_SECRET_KEY="dummy<any non-null string>
export MSDPC_MASTER_SERVER=<primary server>
export MSDPC_ALIAS=<storage_server_name>_test1

/usr/openv/pdde/pdcr/bin/msdpcldutil create -b rv-worm1
-v dv-worm --mode ENTERPRISE --min 1D --max 3D --enable_sas
```

Alternatively, you can provide an access token that you receive from Cohesity to create the WORM bucket or volume. This option is not recommended because the media server must connect to the Recovery Vault web server and Cohesity has to provide the Recovery Vault web server URI.

- `MSDPC_RVLT_API_URI` - A new environment parameter for use when Cohesity provides a different endpoint.
- `MSDPC_ACCESS_TOKEN` - An access token which is part of the credentials that Cohesity provides.
- `MSDPC_CMS_CRED_NAME` - The credential name provided for storing the credentials.

  Example output

  ```
  export MSDPC_CMS_CRED_NAME=your_cms_credential_name
  export MSDPC_ALIAS=your_alias_name
  export MSDPC_REGION=your_region
  export MSDPC_PROVIDER=vazure
  export MSDPC_ENDPOINT="https://your_storage_account.blob.core.windows.net/
  export MSDPC_MASTER_SERVER=<primary server>
  ```

Configuring FortKnox for NetBackup Amazon and Amazon Government

Use the following procedure to configure FortKnox for NetBackup for Amazon and Amazon Government.

Note: FortKnox supports multiple options. For FortKnox Amazon and Amazon Government options in the web UI, you must contact your Cohesity NetBackup account manager for credentials or with any questions.

Table 7-7

Steps	Task	Instructions
Step 1	Retrieve credentials	Retrieve FortKnox credentials from your Cohesity NetBackup account manager
Step 2	(Optional) Create an MSDP storage server if one does not exist.	See *Configuring MSDP server-side deduplication* in the *NetBackup Deduplication Guide* for more information.
Step 3	Add a disk pool.	In the NetBackup web UI, create a disk pool. Follow the procedure in *Create a disk pool* in the *NetBackup Web UI Administrator's Guide*. ■ In the **Volumes** step: ■ Select the option of **FortKnox Amazon** or **FortKnox Amazon Government** from the **Cloud storage provider** drop-down. ■ Select the appropriate region. ■ In the **Associate credentials** section, select **Add a New Credential** and enter a **Refresh token** provided by the provisioning team. Or, you can use **Select existing credentials** if credentials exist for Amazon. The **Credential name** must use alphanumeric characters with hyphens or underscores and cannot contain spaces or illegal characters ■ Enter the cloud bucket name for which you have provided the credentials. FortKnox Amazon provides the following two storage classes: ■ FortKnox AWS Warm Vault ■ FortKnox AWS Cold Vault

Table 7-7 *(continued)*

Steps	Task	Instructions
Step 4	Add a storage unit.	In the NetBackup web UI, create a storage unit. Follow the procedure in *Create a storage unit* in the *NetBackup Web UI Administrator's Guide*. When you create the storage unit, select the **Media Server Deduplication Pool (MSDP)** option. In the **Disk pool** step, select the disk pool that was created in Step 3.

Note: Use the web UI and update the refresh token within **Credential management**.

Configuring FortKnox for NetBackup Amazon and Amazon Government using the CLI

Use the following procedure to configure FortKnox for NetBackup for Amazon and Amazon Government using the CLI.

Table 7-8 Steps for configuring FortKnox for Amazon and Amazon Government with the CLI

Steps	Task	Instructions
Step 1	Retrieve credentials.	Retrieve FortKnox credentials from your Cohesity NetBackup account manager.

Table 7-8 Steps for configuring FortKnox for Amazon and Amazon Government with the CLI *(continued)*

Steps	Task	Instructions
Step 2	Add credentials using the **Credential management** option.	Log into NetBackup web UI and perform the following: **1** On the left, click **Credential management**. **2** On the **Named credentials** tab, click **Add** and provide the following properties: ■ **Credential name** The **Credential name** must use alphanumeric characters with hyphens or underscores and cannot contain spaces or illegal characters. ■ **Tag** ■ **Description** **3** Click **Next**. **4** In the drop-down, select **FortKnox**. **5** Click **FortKnox Amazon**. **6** Add the **Refresh token**. **7** Select or add a role that can access this credential. **8** Review the information and click **Finish**.
Step 3	Create an MSDP storage server.	See "Configuring media server deduplication in NetBackup" on page 63.
Step 4	Create a cloud instance alias.	Use the following examples depending on your environment: ■ Creating an FortKnox Amazon cloud instance alias: `/usr/openv/netbackup/bin/admincmd/csconfig cldinstance -as -in FortKnox-Amazon -sts <storage server> -lsu_name <lsu name>` The cloud alias name is **<storage server>_<lsu name>**, and is used to create a bucket.

Table 7-8 Steps for configuring FortKnox for Amazon and Amazon Government with the CLI *(continued)*

Steps	Task	Instructions
Step 6	Create a configuration file, then run `nbdevconfig` command.	

Table 7-8 Steps for configuring FortKnox for Amazon and Amazon Government with the CLI *(continued)*

Steps	Task	Instructions
		Configuration file content for adding a new cloud LSU (configuration setting and description): ■ V7.5 "operation" "add-lsu-cloud" string - Specifies the value "add-lsu-cloud" for adding a new cloud LSU. ■ V7.5 "lsuName" " " string - Specifies the LSU name. ■ V7.5 "cmsCredName" " " string - Specifies the **Credential name** created using credential management. ■ V7.5 "lsuCloudBucketName" " " string - Specifies the cloud bucket name. ■ V7.5 "lsuCloudBucketSubName" " " string - Multiple cloud LSUs can use the same cloud bucket. This value distinguishes different cloud LSUs. ■ V7.5 "lsuEncryption" " " string - Optional value and the default is NO. Sets the encryption property for the current LSU. ■ V7.5 "lsuKmsEnable" " " string - Optional value and the default is NO. Enables KMS for the current LSU. ■ V7.5 "lsuKmsKeyGroupName" " " string - Optional value. Key group name is shared among all LSUs. Key group name must have valid characters: A-Z, a-z, 0-9, _ (underscore), - (hyphen), : (colon), . (period), and space. ■ V7.5 "lsuKmsServerName" " " string - Optional value. KMS server name is shared among all LSUs. ■ V7.5 "lsuKmsServerType" " " string - Optional value. For an example of a configuration file with encryption disabled, refer to the Azure section. See "Configuring FortKnox for NetBackup Azure and Azure Government using the CLI" on page 365. **Note:** All encrypted LSUs in one storage server must use the same keygroupname and kmsservername. If you use the nbdevconfig command to add a new encrypted cloud LSU and one exists in this MSDP, the keygroupname must be the same as the keygroupname in the previous encrypted LSU. After you create the configuration file, run the nbdevconfig command: /usr/openv/netbackup/bin/admincmd/nbdevconfig -setconfig -storage_server **<storage server>**

Table 7-8 Steps for configuring FortKnox for Amazon and Amazon Government with the CLI *(continued)*

Steps	Task	Instructions
		`-stype PureDisk` `-configlist` **`<configuration file path>`** **Note:** The parameter **<storage server>** must be the same as the parameter **<storage server>** in Step 4.
Step 7	Create disk pool.	Create disk pool by running the `nbdevconfig` command. The following are examples of using the `nbdevconfig` command: Example 1: `/usr/openv/netbackup/bin/admincmd/nbdevconfig` `-previewdv -storage_servers` **`<storage server name>`** `-stype PureDisk` `| grep` **`<LSU name>`** `> /tmp/dvlist` Example 2: `/usr/openv/netbackup/bin/admincmd/nbdevconfig` `-createdp -dp` **`<disk pool name>`** `-stype PureDisk -dvlist /tmp/dvlist` `-storage_servers` **`<storage server name>`** **Note:** You can also create the disk pool from the NetBackup web UI.
Step 8	Create storage unit.	Create storage unit by using `bpstuadd` command. The following are examples of using the `bpstuadd` command: `/usr/openv/netbackup/bin/admincmd/bpstuadd` `-label` **`<storage unit name>`** `-odo 0 -dt 6 -dp` **`<disk pool name>`** `-nodevhost` **Note:** You can also create the storage server from the NetBackup web UI.

Note: Use the web UI and update the refresh token within **Credential management**.

About `csconfig cldinstance` changes for FortKnox for Amazon and Amazon Government

The `csconfig cldinstance` command displays the `Need Token Renew` flag that retrieves alias information (Yes/No). When `Yes`, the FortKnox expects the storage account and refresh token credentials instead of storage account and access key.

The cloud instance has an option to disable (0) or enable (1) the need token renew (-ntr) option which has a default value of yes (1) for `FortKnox-Amazon` and `FortKnox-Amazon-Gov`.

Example usage of `csconfig cldinstance` with `-ntr`:

```
csconfig cldinstance -us -in <instance name> -sts <alias name> -ntr <0,1>
```

Note: When you add the cloud LSU on a back-level media server using the CLI, the `-ntr` option must be set to `No` (0). You must set the option to No because older versions of the media server don't have support for token based credentials. When you use a NetBackup storage server version 10.3.1 or newer, the cloud alias instance must have the `-ntr` option set to `Yes`. The setting cannot be set to `No`.

About `nbcldutil` changes for FortKnox for Amazon and Amazon Government

Starting with NetBackup 10.3.1, the `nbcldutil` command does not support `-validatecreds` options when you configure FortKnox for Amazon and Amazon Government.

About `msdpcldutil` changes for FortKnox for Amazon and Amazon Government

To use this utility, you must create a credential name using the NetBackup web UI and also create a cloud alias using the `csconfig` command similar to the following example:

```
/usr/openv/netbackup/bin/admincmd/csconfig cldinstance
-as -in FortKnox-Amazon -sts <storage_server_name>
-stype PureDisk -lsu_name test1

Successfully added storage server(s): <storage_server_name>_test1
```

Added the `--enable_sts` option to use for FortKnox Amazon. Additionally, if the `--enable _sts` option is used you must export the following environment variables:

- `MSDPC_MASTER_SERVER` - This option is name of the NetBackup primary server.
- `MSDPC_ALIAS` - This option is the cloud alias created using `csconfig`.

- MSDPC_ACCESS_KEY - A credential name and MSDPC_SECRET_KEY is a dummy string.
  In addition to access key and secret key, NetBackup supports the MSDPC_CMS_CRED_NAME variable.
- MSDPC_CMS_CRED_NAME - This is a credential name.

Example output

```
export MSDPC_PROVIDER=vamazon
export MSDPC_REGION="us-east-1"
export MSDPC_CMS_CRED_NAME=<credential name>
export MSDPC_MASTER_SERVER=<primary server>
export MSDPC_ALIAS=<storage_server_name>_testnew

/usr/openv/pdde/pdcr/bin/msdpcldutil create -b rv-worm1
-v dv-worm --mode GOVERNANCE --min 1D --max 3D --enable_sts
```

Alternatively, you can provide an access token that you receive from Cohesity to create the WORM bucket or volume. This option is not recommended because the media server must connect to the Recovery Vault web server and Cohesity has to provide the Recovery Vault web server URI.

- MSDPC_RVLT_API_URI - A new environment parameter for use when Cohesity provides a different endpoint.
- MSDPC_ACCESS_TOKEN - An access token which is part of the credentials that Cohesity provides.

Configuring FortKnox Google for NetBackup

Use the following procedure to configure FortKnox Google for NetBackup.

Note: For FortKnox Google options in the web UI, you must contact your Cohesity NetBackup account manager for credentials or with any questions.

Table 7-9 Steps for configuring FortKnox Google

Steps	Task	Instructions
Step 1	Retrieve credentials	Retrieve FortKnox credentials from your Cohesity NetBackup account manager

Table 7-9 Steps for configuring FortKnox Google *(continued)*

Steps	Task	Instructions
Step 2	(Optional) Create an MSDP storage server if one does not exist.	See *Configuring MSDP server-side deduplication* in the *NetBackup Deduplication Guide* for more information.
Step 3	Add a disk pool.	In the NetBackup web UI, create a disk pool. Follow the procedure in *Create a disk pool* in the *NetBackup Web UI Administrator's Guide*. ■ In the **Volumes** step: ■ Select the option of **FortKnox Google** from the **Cloud storage provider** drop-down. ■ Select the appropriate region. ■ In the **Associate credentials** section, select **Add a New Credential** and enter a **Refresh token** provided by the provisioning team. Or, you can use **Select existing credentials** if credentials exist for Google. The **Credential name** must use alphanumeric characters with hyphens or underscores and cannot contain spaces or illegal characters ■ Enter the cloud bucket name for which you have provided the credentials. FortKnox Google provides the following two storage classes: ■ FortKnox Google Warm Vault ■ FortKnox Google Cold Vault
Step 4	Add a storage unit.	In the NetBackup web UI, create a storage unit. Follow the procedure in *Create a storage unit* in the *NetBackup Web UI Administrator's Guide*. When you create the storage unit, select the **Media Server Deduplication Pool (MSDP)** option. In the **Disk pool** step, select the disk pool that was created in Step 3.

Note: Use the web UI and update the refresh token within **Credential management**.

Configuring FortKnox Google for NetBackup using the CLI

Use the following procedure to configure FortKnox Google for NetBackup using the CLI.

Table 7-10 Steps for configuring FortKnox Google with the CLI

Steps	Task	Instructions
Step 1	Retrieve credentials.	Retrieve FortKnox credentials from your Cohesity NetBackup account manager.
Step 2	Add credentials using the **Credential management** option.	Log into NetBackup web UI and perform the following: **1** On the left, click **Credential management**. **2** On the **Named credentials** tab, click **Add** and provide the following properties: ■ **Credential name** The **Credential name** must use alphanumeric characters with hyphens or underscores and cannot contain spaces or illegal characters. ■ **Tag** ■ **Description** **3** Click **Next**. **4** In the drop-down, select **FortKnox**. **5** Click **FortKnox Google**. **6** Add the **Refresh token**. **7** Select or add a role that can access this credential. **8** Review the information and click **Finish**.
Step 3	Create an MSDP storage server.	See "Configuring media server deduplication in NetBackup" on page 63.

Table 7-10 Steps for configuring FortKnox Google with the CLI *(continued)*

Steps	Task	Instructions
Step 4	Create a cloud instance alias.	Use the following examples depending on your environment: ■ Creating a FortKnox Google cloud instance alias: `/usr/openv/netbackup/bin/admincmd/csconfig cldinstance -as -in FortKnox-Google -sts` **`<storage server>`** `-lsu_name` **`<lsu name>`** ■ Creating an FortKnox Google Cold Vault cloud instance alias `/usr/openv/netbackup/bin/admincmd/csconfig cldinstance -as -in FortKnox-Google -sts <storage server> -lsu_name <lsu name> -storage_tier FortKnox-Google-Cold-Vault` The cloud alias name is **<storage server>_<lsu name>**, and is used to create a bucket.

Table 7-10 Steps for configuring FortKnox Google with the CLI *(continued)*

Steps	Task	Instructions
Step 6	Create a configuration file, then run `nbdevconfig` command.	Configuration file content for adding a new cloud LSU (configuration setting and description): ■ V7.5 `"operation" "add-lsu-cloud"` string - Specifies the value `"add-lsu-cloud"` for adding a new cloud LSU. ■ V7.5 `"lsuName" " "` string - Specifies the LSU name. ■ V7.5 `"cmsCredName" " "` string - Specifies the **Credential name** created using credential management. ■ V7.5 `"lsuCloudBucketName" " "` string - Specifies the cloud bucket name. ■ V7.5 `"lsuCloudBucketSubName" " "` string - Multiple cloud LSUs can use the same cloud bucket. This value distinguishes different cloud LSUs. ■ V7.5 `"lsuEncryption" " "` string - Optional value and the default is `NO`. Sets the encryption property for the current LSU. ■ V7.5 `"lsuKmsEnable" " "` string - Optional value and the default is `NO`. Enables KMS for the current LSU. ■ V7.5 `"lsuKmsKeyGroupName" " "` string - Optional value. Key group name is shared among all LSUs. Key group name must have valid characters: `A-Z, a-z, 0-9, _ (underscore), - (hyphen), : (colon), . (period), and space`. ■ V7.5 `"lsuKmsServerName" " "` string - Optional value. KMS server name is shared among all LSUs. ■ V7.5 `"lsuKmsServerType" " "` string - Optional value. **Note:** All encrypted LSUs in one storage server must use the same `keygroupname` and `kmsservername`. If you use the `nbdevconfig` command to add a new encrypted cloud LSU and one exists in this MSDP, the `keygroupname` must be the same as the `keygroupname` in the previous encrypted LSU. After you create the configuration file, run the `nbdevconfig` command: `/usr/openv/netbackup/bin/admincmd/nbdevconfig -setconfig -storage_server` **`<storage server>`** `-stype PureDisk -configlist` **`<configuration file path>`** **Note:** The parameter **<storage server>** must be the same as the parameter **<storage server>** in Step 4.

Table 7-10 Steps for configuring FortKnox Google with the CLI *(continued)*

Steps	Task	Instructions
Step 7	Create disk pool.	Create disk pool by running the `nbdevconfig` command. The following are examples of using the `nbdevconfig` command: Example 1: `/usr/openv/netbackup/bin/admincmd/nbdevconfig -previewdv -storage_servers` **`<storage server name>`** `-stype PureDisk	grep` **`<LSU name>`** `> /tmp/dvlist` Example 2: `/usr/openv/netbackup/bin/admincmd/nbdevconfig -createdp -dp` **`<disk pool name>`** `-stype PureDisk -dvlist /tmp/dvlist -storage_servers` **`<storage server name>`** **Note:** You can also create the disk pool from the NetBackup web UI.
Step 8	Create storage unit.	Create storage unit by using `bpstuadd` command. The following are examples of using the `bpstuadd` command: `/usr/openv/netbackup/bin/admincmd/bpstuadd -label` **`<storage unit name>`** `-odo 0 -dt 6 -dp` **`<disk pool name>`** `-nodevhost` **Note:** You can also create the storage server from the NetBackup web UI.

Note: Use the web UI and update the refresh token within **Credential management**.

About `msdpcldutil` changes for FortKnox Google

To use this utility, you must create a credential name using the NetBackup web UI and also create a cloud alias using the `csconfig` command similar to the following example:

```
/usr/openv/netbackup/bin/admincmd/csconfig cldinstance
-as -in FortKnox-Google -sts  <storage_server_name>
-stype PureDisk -lsu_name test101
```

```
Successfully added storage server(s): <storage_server_name>_test101
```

Added the `--enable_oauth` option to use for FortKnox Google. Additionally, if the `-enable_oauth` option is used you must export the following environment variables:

- `MSDPC_MASTER_SERVER` - This option is name of the NetBackup primary server.

Example output

```
export MSDPC_PROVIDER=vgoogle
export MSDPC_REGION="US-EAST1"
export MSDPC_ENDPOINT="storage.googleapis.com"
export MSDPC_MASTER_SERVER=<primary server>
export MSDPC_ALIAS=<storage_server_name>_test101
export MSDPC_CMS_CRED_NAME=<cms_cred_name>

/usr/openv/pdde/pdcr/bin/msdpcldutil create -b rv-worm1
-v dv-worm --mode GOVERNANCE --min 1D --max 3D
--live 2023-03-31 --enable_oauth
```

If a cred file is used for the `msdpcldutil` command, do not add any quotes to the values.

Migrating from standard authentication to token-based authentication for Recovery Vault

If you have already configured Veritas Alta Recovery Vault with an older version of NetBackup, you must upgrade to a newer version. To make use of the token-based authentication for enhanced security, you must upgrade the primary and the media server to the following version to use this feature.

- NetBackup 10.2 release for Azure.
- NetBackup 10.4 release for Amazon.

To migrate the credentials

1. Contact NetBackup Technical Support and ask for new credentials.
2. Log into NetBackup web UI and add the new credentials to **Credential management**.
   - On the left, click **Credential management**.
   - On the **Named credentials** tab, click **Add** and provide the following properties:

- Credential name
- Tag
- Description

- Click **Next**.
- In the drop-down, select **FortKnox**.

  Note: For NetBackup 11.0 and earlier releases, select **Veritas Alta Recovery Vault** in the drop-down list.

- Click **FortKnox Azure**, **FortKnox Amazon**, or **FortKnox Google**.

  Note: For NetBackup 11.0 and earlier releases, select **Veritas Alta Recovery Vault Azure** or **Veritas Alta Recovery Vault Amazon** in the drop-down list.

- Add the **Storage account** and **Refresh token**.
- Select or add a role that can access this credential.
- Review the information and click **Finish**.

3 Update the `-ntr` option with the `csconfig cldinstance` command.

Example:

```
/usr/openv/netbackup/bin/admincmd/csconfig cldinstance -us -in
<instance name> -sts <alias name> -ntr 1
```

Confirm the change by making sure that the need token renew option `-ntr` is set to `1` for enabling this option on the storage server:

```
<install path>/netbackup/bin/admincmd/csconfig cldinstance -i
```

4 Update the credentials using `nbdevconfig`.

Create a configuration file with `cmsCredName` as the credential name that you created using the **Credential management**.

Example of the configuration file:

```
V7.5 "operation" "update-lsu-cloud" string
V7.5 "lsuName" "myvolume" string
V7.5 "cmsCredName" "RVLT-creds" string
V7.5 "lsuCloudBucketName" "mybucket" string
V7.5 "lsuCloudBucketSubName" "myvolume" string
```

5 Use the new configuration file to update the credentials.

```
<install path>/netbackup/bin/admincmd/nbdevconfig
-setconfig -stype PureDisk -storage_server <storage_server>
-configlist <config file path>
```

Restart the services on the primary server and the media server for the changes to take effect.

6 Verify the restore of the old backup and run a new backup. Restore the new backup.

About MSDP cloud immutable (WORM) storage support

Cloud immutable storage allows you to store the backup data in the cloud, which you write once but you cannot change or delete it. This feature is supported on Red Hat Enterprise Linux and SUSE Linux Enterprise operating systems only.

This feature does not support legal hold and bucket default retention.

NetBackup supports the following cloud immutable storages:

- Amazon S3 immutable storage
  See “About immutable object support for AWS S3” on page 392.
- Amazon S3 compatible storages
  See “About immutable object support for AWS S3 compatible platforms” on page 396.
- Microsoft Azure immutable storage
  See “About immutable storage support for Azure blob storage ” on page 397.
- Google cloud storage

See “About object-level immutable storage support for Google Cloud Storage” on page 398.

With NetBackup 10.0.1, you can use cloud immutable storage in a cluster environments. For more information, See “About using the cloud immutable storage in a cluster environment” on page 400.

Creating a cloud immutable storage unit using the web UI

Use the NetBackup Web UI to create a cloud immutable storage unit. The following steps describe the process to create a cloud immutable storage unit.

Ensure that the MSDP storage server is created before performing the following steps.

For Azure cloud immutable storage, ensure that a storage account is created with the version-level immutability support enabled.

To create a cloud immutable storage unit

1. On the NetBackup Web UI, navigate to **Storage > Disk pools**, and click **Add**.
2. In **Disk pool options**, click **Change** to select a storage server.

   Enter the **Disk pool name**.

   If **Limit I/O streams** is left cleared, the default value is Unlimited and may cause performance issues.

   After all required information is added, click **Next**.
3. From the **Volume** drop-down list select a volume or add a new volume.

   On the **Cloud storage provider** window, select the provider from the list.

   Under **Region**, select the appropriate region.

   Enter the credentials to complete the setup. You can configure additional options here such as adding a proxy server.

   Under **WORM**, check **Use object lock**. Select the retention mode and enter the lock duration in days or years.

   Under **Cloud bucket**, select **Select or create cloud bucket** and click **Retrieve list**. Select a bucket from the list. You can also provide the bucket name.

   If encryption is needed, select the data encryption option for data compression and encryption. MSDP can use KMS encryption which encrypts the data using a managed key. Using KMS requires that a KMS server has previously been configured.

   Enter all required information based on the selection and click **Next**.
4. In **Replication**, click **Next**.

5 On the **Review** page, verify that all settings and information are correct. Click **Finish**.

The disk pool creation and replication configuration continue in the background if you close the window. If there is an issue with validating the credentials and configuration of the replication, you can use the **Change** option to adjust any settings.

6 In the **Storage unit** tab, click **Add**.

7 Select **Media Server Deduplication Pool (MSDP)** and click **Start**.

8 In **Basic properties**, enter the **Name** of the MSDP storage unit and click **Next**.

9 Select the disk pool that was created and select the **Enable WORM/Lock until expiration** box, and click **Next**.

10 In **Media server**, use the default selection of **Allow NetBackup to automatically select**, and click **Next**.

If it has multiple Media servers, please select the version 9.1 or later.

11 Review the setup of the storage unit and then click **Save**.

Updating a cloud immutable volume

You can update the retention mode and lock duration of the existing cloud immutable volume.

To update a cloud immutable volume

1 On the NetBackup Web UI, navigate to **Storage** > **Disk pools** and click the volume name.

2 Under the **Volume options**, click the actions menu for the volume.

3 Click **Edit retention mode** to update the retention mode of the volume.

In the **Edit retention mode** window, select the retention mode as **Enterprise** or **Compliance**.

Select the existing account credentials or cloud administrator credentials.

Click **Save**.

4 Click **Edit lock duration** to update the lock duration of the volume.

In the **Edit lock duration** window, select minimum and maximum lock duration in days or years.

Select the existing account credentials or cloud administrator credentials.

Click **Save**.

About immutable object support for AWS S3

NetBackup 9.1 and later versions support cloud immutable (WORM) storage with S3 Object Lock. For more information about Amazon S3 Object Lock, see https://docs.aws.amazon.com/AmazonS3/latest/userguide/object-lock-overview.html.

Cloud administrator and backup administrator need specific permissions to configure and use immutable storage. Cloud administrators need a set of permissions to manage the bucket and cloud volume in the cloud and backup administrators need permissions to manage backup data.

See "AWS user permissions to create the cloud immutable volume" on page 393.

Backup images can be locked in one of the following two retention modes:

- Compliance mode
  Users cannot overwrite or delete the data that is protected using the compliance mode for the defined retention period. Once you set a retention period for the data storage, you can extend it but cannot shorten it.
- Enterprise mode
  Users require special permissions to disable the retention lock and then delete the image. Only the cloud administrator user can disable the retention lock and then the delete the image if required. You can use the enterprise mode to test the retention period behavior before you use compliance mode.

Cloud immutable volume (Cloud LSU) is a cloud volume with the following differences than normal cloud volumes:

- The bucket is Object Lock enabled.
- A retention range is defined for the cloud volume. The retention of any backup images must be in this range. NetBackup checks this condition when the backup policy is created.
  You can define and modify this range in the NetBackup web UI.

See "Creating a cloud immutable storage unit using the web UI" on page 390.

See "Updating a cloud immutable volume" on page 391.

See "Extend the cloud immutable volume live duration automatically" on page 393.

See "Performance tuning" on page 393.

See "AWS user permissions to create the cloud immutable volume" on page 393.

See "About bucket policy for immutable storage" on page 395.

Extend the cloud immutable volume live duration automatically

A cloud immutable volume that has an object locking set has an expiration date. After the expiration date, the cloud immutable volume is not protected. However, the live duration of the cloud immutable volume is extended automatically to the maximum value that you have set in the volume configuration.

In the Amazon S3 storage, if you turn on bucket policy, the MSDP cloud does not have permissions to extend the cloud immutable volume live duration and the automatic extension is disabled. In this case, you must use `msdpcldutil` to extend the live duration of the cloud immutable volume manually.

Performance tuning

MSDP spad process has a retention cache. It saves the data container's retention time. When data container's retention time is less than `retentionCacheTimeThreshold`, it does not deduplicate again to quick reclaim the storage. If it has dedupe, the retention time can be extended and cannot be deleted.

The config items are in cloudlsu.cfg,

Parameter	Descripiton	Default value
`retentionCacheSizeThreshold`	Maximum number of data container's retention information is saved in the retention cache. Minimum number saves the memory.	10000000
`retentionCacheTimeThreshold`	When data container retention time is less than this threshold, it does not dedupe again.	432000

AWS user permissions to create the cloud immutable volume

MSDP follows the principle of a least privilege to provision and use S3 immutable storage.

You protect the data with the immutable storage by doing the resource management and using the resources. The resource management tasks such as creating or deleting buckets, enabling Object Lock on buckets are system-level tasks. Using the resource tasks such as running backup or restore jobs, which transfer the data to and from S3 immutable storage are user-level tasks.

These two tasks need different sets of permissions. The principal who has the first set of permissions is a cloud administrator, and the principal who has the second set of permissions is a backup administrator.

Amazon cloud users need the permissions to manage and use the cloud immutable volumes.

Cloud administrator needs the permissions to run `msdpcldutil` to manage cloud volumes.

```
"s3:BypassGovernanceRetention",
"s3:CreateBucket",
"s3:DeleteBucket",
"s3:DeleteObject",
"s3:DeleteObjectVersion",
"s3:GetBucketLocation",
"s3:GetBucketObjectLockConfiguration",
"s3:GetBucketVersioning",
"s3:GetObject",
"s3:GetObjectRetention",
"s3:GetObjectVersion"
"s3:ListAllMyBuckets",
"s3:ListBucket",
"s3:ListBucketVersions",
"s3:PutBucketObjectLockConfiguration",
"s3:PutBucketVersioning",
"s3:PutObject",
"s3:PutObjectRetention",
```

Backup administrator needs the following permissions to configure immutable cloud LSU from Web UI and run data protection jobs such as backup, restore, duplication, replication, and so on.

```
"s3:BypassGovernanceRetention",
"s3:DeleteObject",
"s3:DeleteObjectVersion",
"s3:GetBucketLocation",
"s3:GetBucketObjectLockConfiguration",
"s3:GetBucketVersioning",
"s3:GetObject",
"s3:GetObjectRetention",
"s3:GetObjectVersion",
"s3:ListAllMyBuckets",
"s3:ListBucket",
"s3:ListBucketVersions",
```

```
"s3:PutObject",
"s3:PutObjectRetention",
```

About bucket policy for immutable storage

Bucket policy protects the metadata objects of immutable storage, such as `lockdown-mode.conf` and `lsu-worm.conf` for each volume or sub-bucket. To update the bucket policy, you must run `msdpcldutil update bucket-policy` command.

If the bucket already has some bucket policy, cloud administrator needs to merge the existing bucket policy with the policy for immutable storage manually. For information about editing the S3 bucket policy, see Adding a bucket policy using the Amazon S3 console topic in the AWS documentation.

Following is the example of bucket policy for immutable storage in AWS S3.

```
{
    "Version": "2012-10-17",
    "Id": "vtas-lockdown-mode-file-protection",
    "Statement": [
        {
            "Sid": "vrts-lockdown-file-read-only",
            "Effect": "Deny",
            "Principal": "*",
            "Action": [
                "s3:DeleteObject",
                "s3:PutObject",
                "s3:PutObjectRetention"
            ],
            "Resource": [
                "arn:aws:s3:::bucket-name/volume-name/lockdown-mode.conf",
                "arn:aws:s3:::bucket-name/volume-name/lsu-worm.conf",
                "arn:aws:s3:::bucket-name/volume-name/lockdown-mode.conf",
                "arn:aws:s3:::bucket-name/volume-name/lsu-worm.conf",
                "arn:aws:s3:::bucket-name/volume-name/lockdown-mode.conf",
                "arn:aws:s3:::bucket-name/volume-name/lsu-worm.conf"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:userid": "YOUR-USER-ID-HERE"
                }
            }
        }
```

```
    ]
}
```

See "AWS user permissions to create the cloud immutable volume" on page 393.

About immutable object support for AWS S3 compatible platforms

From NetBackup 10.0 release, the cloud immutable object support for the following S3 compatible platforms is added:

- HCP (Hitachi Content Platform) for Cloud Scale, version 2.3
  - Cloud admin role and backup admin role are combined into a single role.
  - Only compliance mode is supported.
- Nutanix
  - Cloud admin role and backup admin role are combined into a single role.
  - Only compliance mode is supported.
- Cloudian HyperStore, version 7.2
  - Cloud admin role and backup admin role are combined into a single role.
- Seagate Lyve Cloud (public cloud)
  - Cloud admin role and backup admin role are combined into a single role.
- Veritas Access Cloud
  - Cloud admin role and backup admin role are combined into a single role.
  - Only compliance mode is supported.

See "Creating a cloud immutable storage unit using the web UI" on page 390.

From NetBackup 10.1 release, the cloud immutable object support for the following S3 compatible platforms is added:

- Wasabi (Wasabi cloud storage)
  Cloud admin role and backup admin role are combined into a single role.
- Scality RING/ARTESCA
  - Cloud admin role and backup admin role are combined into a single role.
- EMC-ECS (vesion 3.6.2)
  Cloud admin role and backup admin role are combined into a single role.

See "Creating a cloud immutable storage unit using the web UI" on page 390.

From NetBackup 10.1.1 release, the cloud immutable object support for the following S3 compatible platforms is added:

- Quantum ActiveScale
  - Cloud admin role and backup admin role are combined into a single role.
  - Only compliance mode is supported.
- NetApp StorageGRID Webscale – WAN
  - Cloud admin role and backup admin role are combined into a single role.
  - Only compliance mode is supported.

See "Creating a cloud immutable storage unit using the web UI" on page 390.

From NetBackup 10.3 release, the cloud immutable object support for the following S3 compatible platforms is added:

- IBM cloud object storage (iCOS)
  - Cloud admin role and backup admin role are combined into a single role.
  - Only compliance mode is supported
- DataCore Cloud Cluster Storage
  - Cloud admin role and backup admin role are combined into a single role.
  - Only compliance mode is supported
- NetApp StorageGrid LAN
  It is recommended that you use NetAPP StorageGRID version 11.7.0.4 or later. The older versions may have performance issues.
  - Cloud admin role and backup admin role are combined into a single role.
  - Only compliance mode is supported

See "Creating a cloud immutable storage unit using the web UI" on page 390.

See "Updating a cloud immutable volume" on page 391.

About immutable storage support for Azure blob storage

NetBackup 10.0 and later versions support the immutable storage for Azure Blob Storage to store the backup data. For more information about Azure immutable storage, see Store business-critical blob data with immutable storage.

You can use one of the following time-based retention policies for immutable blob data:

- Locked policy
  You cannot overwrite or delete the data that is protected using the locked policy for the defined retention period. Once you set a retention period for the data storage, you can extend it but cannot shorten it.

- Unlocked policy
  You cannot overwrite or delete the data that is protected using the unlocked policy for the defined retention period. Once you set a retention period for the data storage, you can extend, shorten, or delete it.

See "Creating a cloud immutable storage unit using the web UI" on page 390.

See "Updating a cloud immutable volume" on page 391.

About object-level immutable storage support for Google Cloud Storage

NetBackup 10.4 and later versions support the object-level immutable storage for Google Cloud Storage to store the backup data. For more information about Google Cloud Storage, see Object Retention Lock.

Cloud administrator and backup administrator need specific permissions to configure and use immutable storage. Cloud administrators need a set of permissions to manage the bucket and cloud volume in the cloud and backup administrators need permissions to manage backup data.

Backup images can be locked in one of the following two retention modes:

- Compliance mode
  Users cannot overwrite or delete the data that is protected using the compliance mode for the defined retention period. Once you set a retention period for the data storage, you can extend it but cannot shorten it.
- Enterprise mode
  Users require special permissions to disable the retention lock and then delete the image. Only the cloud administrator user can disable the retention lock and then the delete the image if required. You can use the enterprise mode to test the retention period behavior before you use compliance mode.

Cloud immutable volume (Cloud LSU) is a cloud volume with the following differences from the normal cloud volumes:

- The bucket is Object Lock enabled.
- A retention range is defined for the cloud volume. The retention of any backup images must be in this range. NetBackup checks this condition when the backup policy is created. You can define and modify this range in the NetBackup web UI.

NetBackup uses the Google S3 XML API to effectively handle data management in Google Cloud storage. However, the Google S3 XML API lacks the necessary functionality to manage the retrieval and configuration of the bucket default retention of the bucket. So the NetBackup is unable to determine if S3 buckets have a default

retention policy configured, and as a result, it perceives them as non-object-lock buckets. When opting for a bucket with default retention policy in Google Cloud Storage, certain unexpired objects cannot be deleted.

We recommend that you avoid bucket creation in the Google Cloud console. Create all buckets exclusively in the NetBackup web UI. Additionally, identify any buckets with default retention policy in the Google Web Console and refrain from using them in NetBackup.

See “Creating a cloud immutable storage unit using the web UI” on page 390.

See “Updating a cloud immutable volume” on page 391.

See “Extend the cloud immutable volume live duration automatically” on page 393.

See “Performance tuning” on page 393.

Google cloud storage user permissions to create the cloud immutable volume

MSDP follows the principle of the least privilege to provision and use S3 immutable storage.

You protect the data with the immutable storage by doing the resource management and using the resources. The resource management tasks such as creating or deleting buckets, enabling Object Lock on buckets are system-level tasks. Using the resource tasks such as running backup or restore jobs, which transfer the data to and from S3 immutable storage are user-level tasks.

These two tasks need different sets of permissions. The principal who has the first set of permissions is a cloud administrator, and the principal who has the second set of permissions is a backup administrator.

Cloud administrator needs the permissions to make immutable cloud storage configuration to manage cloud volumes.

```
"storage.buckets.create"
"storage.buckets.delete"
"storage.buckets.enableObjectRetention"
"storage.buckets.get"
"storage.buckets.list"
"storage.buckets.update"
"storage.objects.create"
"storage.objects.delete"
"storage.objects.get"
"storage.objects.list"
"storage.objects.overrideUnlockedRetention"
```

```
"storage.objects.setRetention"
"storage.objects.update"
```

Backup administrator needs the following permissions to configure immutable cloud LSU from the web UI and run data protection jobs such as backup, restore, duplication, replication, and so on.

```
"storage.buckets.get"
"storage.buckets.list"
"storage.objects.create"
"storage.objects.delete"
"storage.objects.get"
"storage.objects.list"
"storage.objects.overrideUnlockedRetention"
"storage.objects.setRetention"
"storage.objects.update"
"storage.multipartUploads.create"
"storage.multipartUploads.abort"
"storage.multipartUploads.listParts"
"storage.multipartUploads.list"
```

About using the cloud immutable storage in a cluster environment

Earlier, NetBackup supported the deployment of the cloud immutable storage in a single node. From NetBackup 10.1, NetBackup supports deployment of the cloud immutable storage in the cluster environments such as Azure Kubernetes Service (AKS), Amazon Elastic Kubernetes Service (EKS), and NetBackup Flex Scale.

See “About MSDP cloud immutable (WORM) storage support” on page 389.

You can back up the data to cloud WORM storage to prevent the data from being deleted or overwritten for a fixed amount of time. Currently, MSDP supports the following cloud immutable storages. You can deploy all these cloud immutable storages in the NetBackup cluster environments.

- Amazon S3 Object lock
  See “About immutable object support for AWS S3” on page 392.
- Amazon S3 compatible storage
  See “About immutable object support for AWS S3 compatible platforms” on page 396.
- Azure immutable storage
  See “About immutable storage support for Azure blob storage ” on page 397.

Troubleshooting the errors when disk volume creation using web UI fails

Creating a disk volume with web UI can fail due to incompatible older versions of media or storage servers. Ensure that both the media and storage servers are 10.3 or later.

The following error messages indicate the incompatibility of media or storage servers:

- `One or more invalid arguments`

  This error appears because you entered invalid inputs or media server is incompatible. Verify the inputs and ensure that both the media and storage servers are 10.3 or later.

- `The object "<bucket>/<disk volume name>/lockdown-mode.conf" or the object "<bucket>/<disk volume name>/lsu-worm.conf" does not exist, or neither object "<bucket>/<disk volume name>/lockdown-mode.conf" nor object "<bucket>/<disk volume name>/lsu-worm.conf" exists`

  Storage server has failed to create cloud volume. The disk volume can still be created if a cloud volume is created using `msdpcldutil`. Retention mode and retention range of the volume that set by `msdpcldutil` are retained. The web UI inputs are ignored.

  See *msdpcldutil* topic in the *NetBackup Commands Reference Guide*.

Deleting the immutable image with the enterprise mode

You can delete the immutable image with the enterprise mode retention lock.

To delete the immutable image with the enterprise mode

1. Export the environment variables `MSDPC_ACCESS_KEY` , `MSDPC_SECRET_KEY` of the cloud administrator.

   ```
   export MSDPC_ACCESS_KEY=<your access key id>

   export MSDPC_SECRET_KEY=<your secret key>
   ```

2. Run the following command to find the backup ID and copy number.

   ```
   catdbutil --worm list --allow_worm
   ```

3 Unlock the retention lock.

```
catdbutil --worm disable --backupid ${my_backup_id} --copynum
${my_copy_num} --allow_worm
```

4 Expire the WORM image by using the NetBackup command on the NetBackup primary server.

```
bpexpdate -backupid ${my_backup_id} -d 0 -try_expire_worm_copy
-copy ${my_copy_num}
```

Deleting the S3 object permanently

When you create an immutable bucket, bucket versioning is enabled. It enables you to restore the objects that are accidentally deleted or overwritten. If you delete an object instead of removing it permanently, Immutable S3 Cloud inserts a delete marker, which becomes the current object version.

You can then restore the previous version. If you overwrite an object, it results in a new object version in the bucket. If you want to delete the protected object permanently, you must delete the objects with their versioning.

About MSDP cloud admin tool

You can create, modify, and view WORM parameters through web UI from NetBackup 10.2 and later versions. So, the `msdpcldutil` command-line usage now is reduced.

Most of the tasks are done from the web UI. However, you can perform the following tasks using `msdpcldutil` only.

- `msdpcldutil update inherit`
- `msdpcldutil history list`
- `msdpcldutil history download`
- `msdpcldutil platform checkperm`
- `msdpcldutil bucket`

For more information, see `msdpcldutil` section in the *NetBackup Commands Reference Guide*.

About AWS IAM Role Anywhere support

Using AWS IAM Role Anywhere you can authenticate AWS S3 storage in a secure way using temporary security credentials. You can set up an IAM Role Anywhere

in your AWS account and use the necessary certificate and private key to authenticate.

Note: IAM Role Anywhere is not supported for FortKnox AWS. Also, switching between IAM Role Anywhere and other authentication types such as Access Key is not supported.

Prerequisites for AWS IAM Role Anywhere configuration

- You must have permissions to create an IAM role anywhere in the AWS account and use the necessary certificate and private key to authenticate.
- Create the following and keep the information handy, before starting with the configuration:
  - Profile ARN
  - Role ARN
  - Trust Anchor ARN
  - IAM Role Anywhere CA Certificate
  - IAM Role Anywhere Private Key
- For detailed information about setting up the resources in AWS, see: https://docs.aws.amazon.com/rolesanywhere/latest/userguide/getting-started.html

Configure IAM Role Anywhere in AWS

Follow these steps to set up IAM Role Anywhere in AWS:

- Create the required certificates
- Create the trust anchor
- Create policy
- Create role
- Create profile

Create the required certificates

You can create a CA cert through Amazon's private CA authority which is chargeable by Amazon or by setting up a private CA authority and creating a CA certificate, at no cost.

To use an Amazon private CA resource, refer to AWS documentation here:

https://docs.aws.amazon.com/acm-pca/latest/userguide/PcaWelcome.html

To create a CA certificate for free by setting up a private CA authority and creating a CA certificate, see:

https://docs.aws.amazon.com/rolesanywhere/latest/userguide/getting-started.html

Using you your certificate and the key

1. Use the root CA certificate file to create a Trust Anchor in AWS.
2. Use the self signed CA certificate file and the private key created from the root CA certificate in NetBackup for authentication.

For AWS certificate specifications, see:

https://docs.aws.amazon.com/rolesanywhere/latest/userguide/trust-model.html

Signature validation

To authenticate a request for credentials, IAM Roles Anywhere validates the incoming signature by using the signature validation algorithm required by the key type of the certificate, for example RSA or ECDSA. After validating the signature, IAM Roles Anywhere checks that the certificate was issued by a certificate authority configured as a trust anchor in the account using algorithms defined by public key infrastructure X.509 (PKIX) standards.

End entity certificates must satisfy the following constraints to be used for authentication:

- The certificates MUST be *X.509v3*.
- Basic constraints MUST include *CA: false*.
- The key usage MUST include *Digital Signature*.
- The signing algorithm MUST include *SHA256* or stronger. *MD5* and *SHA1* signing algorithms are rejected.

Certificates used as trust anchors must satisfy the same requirements for the signature algorithm, but with the following differences:

- The key usage must include *Certificate Sign*, and may include *CRL Sign*. Certificate Revocation Lists (CRLs) are an optional feature of IAM Roles Anywhere.
- Basic constraints MUST include *CA: true*.

Create the trust anchor

To create the trust anchor, see AWS documentation:

https://docs.aws.amazon.com/rolesanywhere/latest/userguide/getting-started.html

Create policy

Create a policy in your AWS console and grant the following permissions required by NetBackup.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetBucketLocation",
                "s3:ListAllMyBuckets",
                "s3:CreateBucket",
                "s3:ListBucket",
                "s3:GetBucketObjectLockConfiguration",
                "s3:DeleteObject",
                "s3:PutObject",
                "s3:GetObject",
                "s3:GetBucketPolicyStatus",
                "s3:GetObjectRetention",
                "s3:DeleteObjectVersion",
                "s3:GetBucketVersioning",
                "s3:BypassGovernanceRetention",
                "s3:GetBucketPolicy",
                "s3:PutBucketPolicy",
                "s3:PutBucketObjectLockConfiguration",
                "s3:DeleteBucket",
                "s3:DeleteBucketPolicy",
                "s3:PutBucketVersioning",
                "s3:GetObjectVersion",
                "s3:ListBucketVersions",
                "s3:PutObjectRetention",
                "s3:RestoreObject"
            ],
            "Resource": [
                "arn:aws:s3:::*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeImages"
            ],
```

```
            "Resource": [
                "*"
            ]
        }
    ]
}
```

Create role

Create a role in your AWS console. See AWS documentation for the details:

https://docs.aws.amazon.com/rolesanywhere/latest/userguide/getting-started.html#getting-started-step2

Your policy should look like this with the required parameters filled out.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": [
                    "rolesanywhere.amazonaws.com"
                ]
            },
            "Action": [
              "sts:AssumeRole",
              "sts:TagSession",
              "sts:SetSourceIdentity"
            ],
            "Condition": {
              "ArnEquals": {
                "aws:SourceArn": [
                  "arn:aws:rolesanywhere:<REGION>:<ACCOUNT
NUMBER>:trust-anchor/<TRUST ANCHOR ID>"
                ]
              },
              "StringEquals": {
                    "<PRINCIPAL TAG CHECK>"
              }
            }
        }
    ]
}
```

NetBackup does not require the ArnEquals and the Principal Tag StringEquals checks, but they are suggested security constraints.

Create profile

Create a profile in your AWS console. For details, see:

https://docs.aws.amazon.com/rolesanywhere/latest/userguide/getting-started.html

Configure a new disk pool using AWS IAM Anywhere

Follow these steps to create a new disk pool using AWS IAM Anywhere.

Table 7-11

Steps	Task	Instructions
Step 1	(Optional) Create an MSDP storage server if one does not exist.	See *Configuring MSDP server-side deduplication* in the *NetBackup Deduplication Guide* for more information.
Step 2	Add a disk pool.	In the NetBackup web UI, create a disk pool. Follow the procedure in *Create a disk pool* in the *NetBackup Web UI Administrator's Guide*. ■ In the **Volumes** step: ■ Select the option of **Amazon** or **Amazon Government** from the **Cloud storage provider** drop-down. ■ Select the appropriate region. ■ In the **Access details for Amazon account** section, for **Authentication type**, select **IAM Role Anywhere**, and then select **Add a New Credential**. See "Managing credentials for MSDP-C" on page 298. You can also use **Select existing credentials** if credentials exist for Amazon. ■ Enter the cloud bucket name for which you have provided the credentials.
Step 3	Add a storage unit.	In the NetBackup web UI, create a storage unit. Follow the procedure in *Create a storage unit* in the *NetBackup Web UI Administrator's Guide*. When you create the storage unit, select the **Media Server Deduplication Pool (MSDP)** option. In the **Disk pool** step, select the disk pool that was created in Step 3.

About Azure service principal support

Using Azure service principal you can authenticate Azure blob storage in a secure way without the exposure and automated renewal of the secret keys. NetBackup uses the provided storage account, client ID, tenant ID, and secret key to retrieve temporary credentials to access the cloud resources.

Note: Service principal is not supported with FortKnox Azure. Also, switching between Azure service principal and other authentication type is not supported.

Note: Using Azure service principal in MSDP Cloud deployments is not FIPS-compliant.

Prerequisites for Azure service principal configuration

- You must have permissions to set up and use service principal to authenticate. For more information, see Azure documentation: https://learn.microsoft.com/en-us/azure/active-directory/develop/howto-create-service-principal-portal
- When you create a service principal, you need to give it a role which allows it to perform the required API calls in NetBackup. You must create a custom role definition using the following permissions Actions permissions:

```
"Microsoft.Storage/storageAccounts/blobServices/containers/delete"
"Microsoft.Storage/storageAccounts/blobServices/containers/read"
"Microsoft.Storage/storageAccounts/blobServices/containers/write"
"Microsoft.Storage/storageAccounts/blobServices/generateUserDelegationKey/action"
"Microsoft.Storage/storageAccounts/blobServices/read"
"Microsoft.Storage/storageAccounts/read"
```

  Following are the required Data actions permissions:

```
"Microsoft.Storage/storageAccounts/blobServices/containers/blobs/delete"
"Microsoft.Storage/storageAccounts/blobServices/containers/blobs/read"
"Microsoft.Storage/storageAccounts/blobServices/containers/blobs/write"
"Microsoft.Storage/storageAccounts/blobServices/containers/blobs/move/action"
"Microsoft.Storage/storageAccounts/blobServices/containers/blobs/add/action"
"Microsoft.Storage/storageAccounts/blobServices/containers/blobs/deleteBlobVersion/action"
"Microsoft.Storage/storageAccounts/blobServices/containers/blobs/immutableStorage/runAsSuperUser/action"
```

- Create the following and keep the information handy, before starting with the configuration:

- Storage Account
- Client ID
- Tenant ID
- Secret Key

Configure Azure service principal

Follow these steps to set up Azure service principal:

- Create a new custom role
- Create a new service principal

Create a new custom role

To create a custom Azure RBAC role to use with the service principal in NetBackup, see Azure documentation:

https://learn.microsoft.com/en-us/azure/role-based-access-control/custom-roles

Note the following:

- The scope of the role must be at the subscription level. This ensures the service principal works for all storage accounts in the subscription.
- Grant the required **Actions** and **Data actions** permissions.

Create a new service principal

To create a new service principal, see Azure documentation:

https://learn.microsoft.com/en-us/entra/identity-platform/howto-create-service-principal-portal

While creating the service principal, use the application (client) ID, directory (tenant) ID, and secret key that you have created.

Ensure that you assign your custom role to the service principal.

Configure a disk pool using Azure service principal

Follow these steps to create a new disk pool using Azure service principal.

Table 7-12

Steps	Task	Instructions
Step 1	(Optional) Create an MSDP storage server if one does not exist.	See *Configuring MSDP server-side deduplication* in the *NetBackup Deduplication Guide* for more information.
Step 2	Add a disk pool.	In the NetBackup web UI, create a disk pool. Follow the procedure in *Create a disk pool* in the *NetBackup Web UI Administrator's Guide*. ■ In the **Volumes** step: ■ Select the option of **Microsoft Azure** or **Microsoft Azure Government** from the **Cloud storage provider** drop-down. ■ Select the appropriate region. ■ In the **Access details for Azure account** section, For **Authentication type**, select **Service principal**, and then select **Add a New Credential**. See "Managing credentials for MSDP-C" on page 298. You can also use **Select existing credentials** if credentials exist for Azure. ■ Enter the cloud bucket name for which you have provided the credentials.
Step 3	Add a storage unit.	In the NetBackup web UI, create a storage unit. Follow the procedure in *Create a storage unit* in the *NetBackup Web UI Administrator's Guide*. When you create the storage unit, select the **Media Server Deduplication Pool (MSDP)** option. In the **Disk pool** step, select the disk pool that was created in Step 3.

About instant access for object storage

The following table describes the platforms supported by instant access for object storage.

Table 7-13 Supported platforms

Supported platforms	Description
Azure Kubernetes Service (AKS)	This platform is supported and enabled by default.

Table 7-13 Supported platforms *(continued)*

Supported platforms	Description
Amazon Elastic Kubernetes Service (EKS)	This platform is supported and enabled by default.
VM in Azure or AWS (Cloud Build-Your-Own, BYO-In-Cloud)	This platform is supported. You must manually enable this option.
MSDP on-premises (BYO, Flex media server, NetBackup appliance)	This platform is supported. You must manually enable this option. The connection from MSDP to object store must have good network bandwidth and latency.

The instant access for object storage is enabled by default on the AKS/EKS platforms. If instant access is not enabled by default, you must manually perform the following steps to enable it:

1. Add the `instant-access-object-store = 1` option into the */etc/msdp-release* file on storage server.
2. On the primary or media server, run the following commands to verify that the `IA_OBJECT_STORE` name is in the `extendedcapabilities` option.

   Example:

   ```
   nbdevconfig -getconfig -stype PureDisk
   -storage_server your_storage_server_name
   |grep IA_OBJECT_STORE
   ```

3. On the primary or media server, run the following commands to reload the storage server attributes:

   ```
   nbdevconfig -getconfig -stype PureDisk
   -storage_server your_storage_server_name > /tmp/flags
   ```

   ```
   nbdevconfig -setconfig -stype PureDisk
   -storage_server your_storage_server_name -configlist /tmp/flags
   ```

Upgrading to NetBackup 10.3 and cluster environment

In the cluster environments such as NetBackup Cloud Scale and NetBackup Flex Scale, you must migrate cloud volume-associated credentials so that you can add the new engines in NetBackup 10.3 or later.

See the *Migrating or upgrading MSDP Cloud and CMS* topic in the *NetBackup Web UI Administration's Guide* to save the cloud credentials in CMS and establish the associate credentials with the existing cloud volumes.

About the cloud direct

The MSDP generates data containers in a storage server and uploads them to the cloud storage for backup. It transmits the coming data stream from the client to the media server and storage server. For restore, the storage server downloads the data container from the cloud storage, reads the data segment from the data container file, and sends it to the client.

The `mtstrmd` service running on the client or the media server writes and reads the image data to or from the cloud storage directly. Only metadata is sent to the storage server. It leverages client resource and reduces the storage and the media server data transmission overhead.

Configuring the cloud direct for backup

A client direct backup must be enabled to use this feature. You can configure cloud direct using the `pdagutil` command. The cloud direct backup is supported on Red Hat Linux only.

Before you configure cloud direct for backup, enable client-side deduplication.

To configure the cloud direct for backup

1. Configure the cloud direct for backup.

   ```
   /usr/openv/pdde/pdag/bin/pdagutil --config-cloud-direct-backup
   --storage-path <storage path> --cache-size <cache size>
   ```

2. Enable the cloud direct for backup.

   ```
   /usr/openv/pdde/pdag/bin/pdagutil --update-cloud-direct-backup
   --enable
   ```

3 Disable cloud direct for backup.

```
/usr/openv/pdde/pdag/bin/pdagutil --update-cloud-direct-backup
--disable
```

4 Update the cache size.

```
/usr/openv/pdde/pdag/bin/pdagutil --update-cloud-direct-backup
--cache-size <cachesize>
```

5 Restart the `mtstrmd` service for the configuration to take effect.

```
/usr/openv/pdde/pdag/bin/mtstrmctl --shut-down
```

Configuring the cloud direct for restore

A client-direct restore for the client must be enabled. You can configure cloud direct using the `pdagutil` command. The cloud direct restore is supported on Red Hat Linux only.

Before you configure cloud direct for restore, enable client-direct restore.

See " Configuring MSDP restores directly to a client" on page 565.

To configure the cloud direct for restore

1 Configure the cloud direct for restore.

```
/usr/openv/pdde/pdag/bin/pdagutil --config-cloud-direct-restore
--storage-path <storage path> --cache-size <cache size>
```

2 Enable the cloud direct for restore after configuration.

```
/usr/openv/pdde/pdag/bin/pdagutil --update-cloud-direct-restore
--enable
```

3 Disable the cloud direct for restore after configuration.

```
/usr/openv/pdde/pdag/bin/pdagutil --update-cloud-direct-restore
--disable
```

4 Update cache size.

```
/usr/openv/pdde/pdag/bin/pdagutil --update-cloud-direct-restore
--cache-size <cachesize>
```

5 Restart the `mtstrmd` service for the configuration to take effect.

```
/usr/openv/pdde/pdag/bin/mtstrmctl --shut-down
```

About MSDP lazy delete

MSDP lazy delete lets you retain the uploaded data to the cloud for the failed jobs for a certain period. This allows the retrying jobs to reference the already successfully uploaded data, reducing data duplication, improving efficiency, and lowering storage costs. For example, if 80% of the data was uploaded before a failure, the retried job uses that 80% and uploads only the remaining 20%. This feature is especially useful with the archive tiers like AWS Glacier or Azure Archive, which have minimum retention requirements.

The MSDP lazy delete is enabled by default. It retains the metadata of failed jobs for 2 days by default. You can update it or disable it by using the `msdpcldutil lazy-delete` command. For more information about the command usage, see the *NetBackup Commands Reference Guide*.

When MSDP lazy delete is disabled, NetBackup image cleanup removes the fragments of the image uploaded to the cloud. The MSDP does not use the unique data of a deleted image even if the data is still stored in the object storage. When the job is run again, it stores the same data in the cloud again. The duplicate data increases in object storage with each retried job.

About verification of the images stored in the archive tier

You can verify the backup images that are stored in archive tiers, including Amazon Glacier, Amazon Glacier Deep Archive, and Azure Archive.

Image verification reads archived fragments and therefore requires warming the data, similar to a restore operation. The same retrieval methods that are used for restore are supported during the verification. For information about restore, see *About restoring data from Amazon Glacier* and *Considerations with Restore of Image Fragments* topics in *NetBackup Cloud Administrator's Guide.*

NetBackup uses the bulk retrieval method by default. To use a different retrieval method, you can create a `GLACIER_RETRIEVAL` touch file on the storage server or media server in the `/usr/openv/netbackup/bin` directory and specify one of the supported values: `bulk`, `standard`, or `expedited`.

For Amazon Glacier, all three retrieval methods are supported, while Amazon Glacier Deep Archive supports only bulk and standard. If the touch file is not present or no value is specified, NetBackup uses the default bulk retrieval method.

Chapter 8

S3 Interface for MSDP

This chapter includes the following topics:

- About S3 interface for MSDP
- Prerequisites for MSDP build-your-own (BYO) server
- Configuring S3 interface for MSDP on MSDP build-your-own (BYO) server
- Identity and Access Management (IAM) for S3 interface for MSDP
- S3 Object Lock In Flex WORM
- S3 APIs for S3 interface for MSDP
- Creating a protection policy for the MSDP object store
- About MSDP S3 object import
- Recovering the MSDP object store data from the backup images
- Instant access for MSDP object store
- Disaster recovery in S3 interface for MSDP
- Limitations in S3 interface for MSDP
- Logging and troubleshooting
- Best practices

About S3 interface for MSDP

S3 is a popular storage interface in the cloud. It can seamlessly work with the cloud native applications. S3 interface for MSDP provides S3 APIs in MSDP server. The S3 interface for MSDP is compatible with Amazon S3 cloud Storage service. It

supports most of the commonly used S3 APIs such as create bucket, delete bucket, store object, retrieve object, list object, delete object, multiipart upload, and so on.

S3 interface for MSDP also supports object versioning, IAM, and identity-based policy. It uses snowball-auto-extract to support small objects batch upload.

S3 interface for MSDP can be configured on MSDP build-your-own (BYO) server, Flex appliance, Flex WORM, and NetBackup on AKS/EKS.

For S3 interface for MSDP configuration on Flex appliance, login to the Flex media server and See "Configuring S3 interface for MSDP on MSDP build-your-own (BYO) server" on page 417.

For S3 interface for MSDP configuration on Flex WORM, See "Managing S3 service from the deduplication shell" on page 755.

For S3 interface for MSDP configuration on NetBackup on AKS/EKS, see the *Using S3 service in MSDP Scaleout* topic of *NetBackup Deployment Guide for Kubernetes Clusters* document.

Note: The time between the clients and the S3 server must be synchronized to have successful API calls.

Prerequisites for MSDP build-your-own (BYO) server

Following are the prerequisites to configure S3 interface for MSDP:

- The storage server operating system must be Red Hat Enterprise Linux.

  Note: S3 interface for MSDP is not supported on SUSE Linux Enterprise.

- It's recommended that the storage server has more than 64 GB of memory and 8 CPUs.
- Ensure that NGINX is installed in the storage server.
  - The NGINX version must be same as the one in the corresponding official RHEL version release. Install it from the corresponding RHEL yum source.
  - Run the following command to confirm that the NGINX is ready:
    `systemctl is-active <ngnix service name>`
  - Configuration under `/etc/<nginx service>/conf.d` must be included in the NGINX configuration.

- If SE Linux is enabled, ensure that the **policycoreutils** and **policycoreutils-python** (for RHEL 7) or **policycoreutils-python-utils** (for RHEL 8) packages are installed from the same RHEL yum source (RHEL server).
  Run the following command to allow S3 interface for MSDP to listen on special port:
  ```
  semanage port -a -t http_port_t -p tcp <nginx port>
  ```
  Run the following command to allow S3 interface for MSDP to connect to the network:
  ```
  setsebool -P httpd_can_network_connect 1
  ```

Configuring S3 interface for MSDP on MSDP build-your-own (BYO) server

After MSDP is configured, you can configure S3 interface for MSDP using the NetBackup web UI or using `s3srv_config.sh` script.

To configure S3 interface for MSDP using the NetBackup web UI, see *Configuring an MSDP object store* topic of the *NetBackup Web UI Administrator's Guide*.

To configure S3 server using `s3srv_config.sh` script

- If you want to use NBCA or ECA type certificates in S3 interface for MSDP, run the following command:

  ```
  /usr/openv/pdde/vxs3/cfg/script/s3srv_config.sh --catype=<type>
  [--port=<port>] [--loglevel=<0-4>]
  ```

  If you want to use your certificates in S3 interface for MSDP, run the following command:

  ```
  /usr/openv/pdde/vxs3/cfg/script/s3srv_config.sh --cert=<certfile>
  --key=<keypath> [--port=<port>] [--loglevel=<0-4>]
  ```

`--catype=<type>`	Certificate Authority type. NBCA: 1 or ECA: 2.
`--cert=<certfile>`	Certificate file for HTTPS.
`--key=<keypath>`	Private key for HTTPS.
`--port=<port>`	S3 server port. Default port is 8443.

`--loglevel=<0-4>`	S3 server log level. ■ None: 0 ■ Error: 1 ■ Warning: 2 ■ Info: 3 (default) ■ Debug: 4
`--help|-h`	Print the usage.

- S3 service is HTTPS service. Default port is 8443.
- This script can run with the root user directly. With other service user, run this command in the following format:
  `sudo -E /usr/openv/pdde/pdcr/bin/msdpcmdrun /usr/openv/pdde/vxs3/cfg/script/s3srv_config.sh <arguments>`
- If multiple certificates exist under `/usr/openv/var/vxss/credentials`, you may see the following configuration error:
  `Too many ca files under /usr/openv/var/vxss/credentials/keystore`
  You can use option `--cert` and `--key` to specify which certificate is used.
- You can enable HTTPS with the certificate, which is not signed by Certificate Authority in S3 interface for MSDP. If S3 interface for MSDP is configured with NBCA as SSL certificate, CA certificate is `/usr/openv/var/webtruststore/cacert.pem` under S3 server host. When you use AWS CLI to connect S3 interface for MSDP, there are two options `--ca-bundle` and `--no-verify-ssl`. Option `--ca-bundle` verifies SSL certificates with corresponding CA certificate bundle. Option `--no-verify-ssl` overrides verifying SSL certificates in AWS CLI command. You can ignore the following warning message.
  `urllib3/connectionpool.py:1043: InsecureRequestWarning: Unverified HTTPS request is being made to host 'xxxx.xxxx.com'. Adding certificate verification is strongly advised. See: https://urllib3.readthedocs.io/en/1.26.x/advanced-usage.html#ssl-warnings`
- Only PEM format of certificate and secret key is supported. Please convert other format of certificate and secret key to PEM format.
- After configuring S3 server, you can check S3 server status.
  Root user: `systemctl status pdde-s3srv`
  Other service users: `sudo -E /usr/openv/pdde/pdcr/bin/msdpcmdrun /usr/openv/pdde/vxs3/cfg/script/s3srv_adm.sh status`
- After configuring S3 server, you can stop or start S3 server.
  Root user: `systemctl stop/start pdde-s3srv`

Other service users: `sudo -E /usr/openv/pdde/pdcr/bin/msdpcmdrun /usr/openv/pdde/vxs3/cfg/script/s3srv_adm.sh stop|start`

- NGINX configurations about S3 server are saved at `/etc/<nginx path>/conf.d/s3srvbyo.conf` and `/etc/<nginx path>/locations/s3srv.conf`. If you have modified the configuration files, you must modify them again after the upgrade.

Changing the certificate in S3 server

S3 server HTTPS certificate must be renewed manually when it expires. Alternatively, you can change NBCA to ECA.

To change NBCA to ECA.

- Run the following command:

```
s3srv_config.sh --changeca --catype=<type>
```

```
s3srv_config.sh --changeca --cert=<certfile> --key=<keypath>
```

Changing the ETAG type of the S3 objects

MSDP S3 server returns an ETAG for each object put in the server. The default ETAG type is SHA256. You can change the ETAG type to MD5 if it is necessary for some S3 clients.

To change the ETAG type the S3 objects

1. Open the S3 server configuration file `<storage>/etc/puredisk/s3srv.cfg`.
2. Edit the value of `EtagType`.

```
; Etag type. Valid values are: SHA256, MD5, DOFP
; Note: MD5 cannot be used with FIPS mode
; Note: DOFP uses MSDP DO finger print as ETAG value. Use this value for
; @restart
EtagType=SHA256
```

3. Restart the S3 server.

```
systemctl restart pdde-s3srv
```

Identity and Access Management (IAM) for S3 interface for MSDP

S3 Identity and Access Management (IAM) helps you control access to the S3 server.

Signing IAM and S3 API requests

MSDP S3 server uses the same signing methods as AWS. Both signature version 4 and version 2 are supported. For more information about signing a request, see the following pages:

- Signature Version 4 signing process
- Signature Version 2 signing process

IAM workflow

In this section, the typical workflow of IAM is described. You can install AWS CLI to send IAM-related API request to complete the tasks.

IAM workflow

1. Reset and get S3 server root user's credentials.

   Create root user credentials. You can use the root user to create users with limited permissions.

   After S3 interface for MSDP is configured, run the following command to create root user's credentials:

   ```
   /usr/openv/pdde/vxs3/cfg/script/s3srv_config.sh --reset-iam-root
   ```

   You can also use this command if you have lost root user's access keys. The new access key and secret key of root user is available in the command output.

   To create or reset root user's credentials using NetBackup web UI, see *Resetting the MSDP object store root credentials* topic of the *NetBackup Web UI Administrator's Guide*.

2. Create a user.

   ```
   aws --endpoint https://<MSDP_HOSTNAME>:8443 [--ca-bundle
   <CA_BUNDLE_FILE>] iam create-user --user-name <USER_NAME>
   ```

3 Attach one or more policies to a user.

```
aws --endpoint https://<MSDP_HOSTNAME>:8443 [--ca-bundle
<CA_BUNDLE_FILE>] iam put-user-policy --user-name <USER_NAME>
--policy-name <POLICY_NAME> --policy-document
file://<POLICY_DOCUMENT_FILE_PATH>
```

4 Create access key for a user.

```
aws --endpoint https://<MSDP_HOSTNAME>:8443 [--ca-bundle
<CA_BUNDLE_FILE>] iam create-access-key [--user-name <USER_NAME>]
```

Note: If you omit the `--user-name` option, the access key is created under the user who sends the request.

5 Delete access key for a user.

```
aws --endpoint https://<MSDP_HOSTNAME>:8443 [--ca-bundle
<CA_BUNDLE_FILE>] iam delete-access-key [--user-name <USER_NAME>]
--access-key-id <ACCESS_KEY>
```

Note: If you omit the `--user-name` option, the access key is deleted under the user who sends the request. You cannot delete the last active access key of a root user.

6 List access keys for a user.

```
aws --endpoint https://<MSDP_HOSTNAME>:8443 [--ca-bundle
<CA_BUNDLE_FILE>] iam list-access-keys [--user-name <USER_NAME>]
```

Note: If you omit the `--user-name` option, the access key is listed under the user who sends the request.

7 Update an access key's status for a user.

```
aws --endpoint https://<MSDP_HOSTNAME>:8443 [--ca-bundle
<CA_BUNDLE_FILE>] iam update-access-key [--user-name <USER_NAME>]
--access-key-id <ACCESS_KEY> --status [Active | Inactive]
```

If you omit the `--user-name` option, the access key is updated under the user who sends the request.

The option `--status` must follow **Active** or **Inactive** parameter (case sensitive).

You cannot update the last active access key of root user to **Inactive** status.

8 Get a specific user policy.

```
aws --endpoint https://<MSDP_HOSTNAME>:8443 [--ca-bundle
<CA_BUNDLE_FILE>] iam get-user-policy --user-name <USER_NAME>
--policy-name <POLICY_NAME>
```

9 List all attached policies for a user.

```
aws --endpoint https://<MSDP_HOSTNAME>:8443 [--ca-bundle
<CA_BUNDLE_FILE>] iam list-user-policies --user-name <USER_NAME>
```

10 Delete a user policy.

```
aws --endpoint https://<MSDP_HOSTNAME>:8443 [--ca-bundle
<CA_BUNDLE_FILE>] iam delete-user-policy --user-name <USER_NAME>
--policy-name <POLICY_NAME>
```

11 Get user information.

```
aws --endpoint https://<MSDP_HOSTNAME>:8443 [--ca-bundle
<CA_BUNDLE_FILE>] iam get-user --user-name <USER_NAME>
```

12 List all users.

```
aws --endpoint https://<MSDP_HOSTNAME>:8443 [--ca-bundle
<CA_BUNDLE_FILE>] iam list-users
```

13 Delete a user.

```
aws --endpoint https://<MSDP_HOSTNAME>:8443 [--ca-bundle
<CA_BUNDLE_FILE>] iam delete-user --user-name <USER_NAME>
```

Note: Before you delete a user, you must delete the user policies and access keys that are attached to the user. You cannot delete a root user.

IAM APIs for S3 interface for MSDP

The Identity and Access Management (IAM) is a web service for securely controlling access to MSDP S3 interface. With IAM, you can centrally manage users, security credentials such as access keys, and permissions that control which resources users can access.

The IAM-related APIs of MSDP S3 interface support only HTTP POST method for all IAM actions.

Common Parameters

The following table contains the parameters that all actions use for signing Signature Version 4 requests with a query string.

Table 8-1 Common parameters

Parameters	Description
`Action`	The action to be performed. Type: string Required: Yes
`Version`	The API version that the request is written for, expressed in the format YYYY-MM-DD. Type: string Required: No
`X-Amz-Algorithm`	The credential scope value that includes your access key, the date, the region, the service, and a termination string. The value is configured in the following format: `access_key/YYYYMMDD/region/service/aws4_request`. For more information, see Task 2: Create a String to Sign for Signature Version 4. Condition: Specify this parameter when you include authentication information in a query string instead of the HTTP authorization header. Type: string Required: Conditional

Table 8-1 Common parameters *(continued)*

Parameters	Description
`X-Amz-Credential`	The credential scope value that includes your access key, the date, the region, the service, and a termination string. The value is configured in the following format: `access_key/YYYYMMDD/region/service/aws4_request`. For more information, see Task 2: Create a String to Sign for Signature Version 4 Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header. Type: string Required: Conditional
`X-Amz-Date`	The date that is used to create the signature. The format must be ISO 8601 basic format (YYYYMMDD'T'HHMMSS'Z'). For example, the following date time is a valid `X-Amz-Date` value: 20220525T120000Z. Condition: `X-Amz-Date` is optional for all requests; it can be used to override the date that is used for signing requests. If the Date header is specified in the ISO 8601 basic format, `X-Amz-Date` is not required. When `X-Amz-Date` is used, it always overrides the value of the Date header. For more information, see Handling Dates in Signature Version 4 Type: string Required: Conditional
`X-Amz-Signature`	Specifies the hex-encoded signature that was calculated from the string to sign and the derived signing key. Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header. Type: string Required: Conditional

Table 8-1 Common parameters *(continued)*

Parameters	Description
`X-Amz-SignedHeaders`	Specifies all the HTTP headers that were included as part of the canonical request. For more information about specifying signed headers, see Task 1: Create a Canonical Request For Signature Version 4 in the Amazon Web Services General Reference. Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header. Type: string Required: Conditional

Common Error Codes

The following error codes are common to IAM APIs. For errors specific to an API action is described in the IAM API section.

Table 8-2 Common error codes

Error code	Description
InvalidClientTokenId	The access key ID provided does not exist in our records. HTTP Status Code: 403
SignatureDoesNotMatch	The request signature we calculated does not match the signature you provided. HTTP Status Code: 403
ValidationError	The input fails to satisfy the constraints that are specified by an AWS service. HTTP Status Code: 400
AccessDeniedException	You do not have sufficient access to perform this action. HTTP Status Code: 400
MissingAction	The request is missing an action or a required parameter. HTTP Status Code: 400

Table 8-2 Common error codes *(continued)*

Error code	Description
NotImplemented	A header you provided implies the functionality that is not implemented. HTTP Status Code: 501

CreateUser

Creates a new IAM user for MSDP S3.

Request Parameters

For information about the parameters that are common to all actions, See "Common Parameters" on page 423.

- `UserName`
  The name of the user to create.
  IAM user names must be unique. User names are case-sensitive.
  Type: String
  Length Constraints: Minimum length of 1. Maximum length of 64.
  Pattern: [\w+=,.@-]+
  Required: Yes

Response Elements

The following element is returned by server.

- `User`
  A structure with details about the new IAM user.
  Type: User object

Errors

For information about the errors that are common to all actions, See "Common Error Codes" on page 425.

- `EntityAlreadyExists`
  The request was rejected because it attempted to create a resource that already exists.
  HTTP Status Code: 409
- `InvalidInput`
  The request was rejected because an invalid or out-of-range value was supplied for an input parameter.
  HTTP Status Code: 400
- `ServiceFailure`

The request processing has failed because of an unknown error, exception, or failure.

HTTP Status Code: 500

Examples

Sample Request:

```
https://msdps3.veritas.com:8443/?Action=CreateUser
&UserName=User1
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<CreateUserResponse
  xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
  <ResponseMetadata>
    <RequestId>16481914289311827O3</RequestId>
  </ResponseMetadata>
  <CreateUserResult>
    <User>
      <CreateDate>2022-03-25T06:57:08Z</CreateDate>
      <UserName>User1</UserName>
    </User>
  </CreateUserResult>
</CreateUserResponse>
```

GetUser

Retrieves the information about the specified IAM user.

If you do not specify a user name, IAM determines the user name implicitly based on the MSDP S3 access key ID used to sign the request to this operation.

Request Parameters

For information about the parameters that are common to all actions, See "Common Parameters" on page 423.

- `UserName`

  The name of the user to get information about.

  This parameter is optional. If it is not included, it defaults to the user making the request.

  Type: String

  Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: [\w+=,.@-]+
Required: No

Response Elements

The following element is returned by server.

- `User`
  A structure with details about the new IAM user.
  Type: User object

Errors

For information about the errors that are common to all actions, See "Common Error Codes" on page 425.

- `NoSuchEntity`
  The request was rejected because it referenced a resource entity that does not exist. The error message describes the resource.
  HTTP Status Code: 404
- `ServiceFailure`
  The request processing has failed because of an unknown error, exception, or failure.
  HTTP Status Code: 500

Examples

Sample Request:

```
https://msdps3.veritas.com:8443/?Action=GetUser
&UserName=User1
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<GetUserResponse
  xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
  <ResponseMetadata>
    <RequestId>16481914289311827O3</RequestId>
  </ResponseMetadata>
  <GetUserResult>
    <User>
      <CreateDate>2022-03-25T06:57:08Z</CreateDate>
      <UserName>User1</UserName>
    </User>
```

```
  </GetUserResult>
</GetUserResponse>
```

ListUsers

Lists all the IAM users of the server.

Request Parameters

For information about the parameters that are common to all actions, See "Common Parameters" on page 423.

This API does not need any specific request parameters.

Response Elements

The following elements are returned by server.

- `Users.member.N`
  A list of users.
  Type: Array of User objects
- `IsTruncated`
  A flag that indicates whether there are more items to return.
  Type: Boolean

Errors

For information about the errors that are common to all actions, See "Common Error Codes" on page 425.

- `ServiceFailure`
  The request processing has failed because of an unknown error, exception, or failure.
  HTTP Status Code: 500

Examples

Sample Request:

```
https://msdps3.veritas.com:8443/?Action=ListUsers
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<ListUsersResponse
   xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
  <ResponseMetadata>
    <RequestId>164820360490589306 9</RequestId>
```

```
  </ResponseMetadata>
  <ListUsersResult>
    <Users>
      <member>
        <CreateDate>2022-03-22T13:35:03Z</CreateDate>
        <UserName>root</UserName>
      </member>
      <member>
        <CreateDate>2022-03-25T06:57:08Z</CreateDate>
        <UserName>User1</UserName>
      </member>
    </Users>
    <IsTruncated>false</IsTruncated>
  </ListUsersResult>
</ListUsersResponse>
```

DeleteUser

Deletes the specified IAM user.

You must delete the items (for example, access keys, policies) that are attached to the user manually before you delete the user.

Request Parameters

For information about the parameters that are common to all actions, See “Common Parameters” on page 423.

- `UserName`

  The name of the user to delete.

  Type: String

  Length Constraints: Minimum length of 1. Maximum length of 64.

  Pattern: [\w+=,.@-]+

  Required: Yes

Errors

For information about the errors that are common to all actions, See “Common Error Codes” on page 425.

- `DeleteConflict`

  The request was rejected because it attempted to delete a resource that has attached subordinate entities. The error message describes these entities.

  HTTP Status Code: 409

- `NoSuchEntity`

The request was rejected because it referenced a resource entity that does not exist. The error message describes the resource.
HTTP Status Code: 404

- `ServiceFailure`
  The request processing has failed because of an unknown error, exception, or failure.
  HTTP Status Code: 500

Examples

Sample Request:

```
https://msdps3.veritas.com:8443/?Action=DeleteUser
&UserName=User1
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<GetUserResponse
  xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
  <ResponseMetadata>
    <RequestId>1648214899748730966</RequestId>
  </ResponseMetadata>
</DeleteUserResponse>
```

CreateAccessKey

Creates a new AWS secret access key and corresponding MSDP S3 access key ID for the specified user. The default status for new keys is Active.

If you do not specify a user name, IAM determines the user name implicitly based on the MSDP S3 access key ID signing the request.

A user can have up to two access keys.

Request Parameters

For information about the parameters that are common to all actions, See "Common Parameters" on page 423.

- `UserName`
  The name of the IAM user that the new key will belong to.
  This parameter is optional. If it is not included, it defaults to the user making the request.
  Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `[\w+=,.@-]+`

Required: No

Response Elements

The following element is returned by server.

- `AccessKey`

  A structure with details about the access key.

  Type: Access key object See "Data Types" on page 444.

Errors

For information about the errors that are common to all actions, See "Common Error Codes" on page 425.

- `LimitExceeded`

  The request was rejected because it attempted to create resources beyond the limits. The error message describes the limit exceeded.

  HTTP Status Code: 409

- `NoSuchEntity`

  The request was rejected because it referenced a resource entity that does not exist. The error message describes the resource.

  HTTP Status Code: 404

- `ServiceFailure`

  The request processing has failed because of an unknown error, exception, or failure.

  HTTP Status Code: 500

Examples

Sample Request:

```
https://msdps3.veritas.com:8443/?Action=CreateAccessKey
&UserName=User1
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<CreateAccessKeyResponse
  xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
  <ResponseMetadata>
    <RequestId>1648431826555152698</RequestId>
  </ResponseMetadata>
```

```
  <CreateAccessKeyResult>
    <AccessKey>
      <AccessKeyId>2PPM4XHAKMG5JHZIUPEUG</AccessKeyId>
      <CreateDate>2022-03-28T01:43:46Z</CreateDate>
      <SecretAccessKey>9TvXcpw2YRYRZXZCyrCELGVWMNBZyJYY95jhDc1xgH
      </SecretAccessKey>
      <Status>Active</Status>
      <UserName>User1</UserName>
    </AccessKey>
  </CreateAccessKeyResult>
</CreateAccessKeyResponse>
```

ListAccessKeys

Returns the information about the access key IDs associated with the specified IAM user. If there is none, the operation returns an empty list.

If the `UserName` field is not specified, the user name is determined implicitly based on the MSDP S3 access key ID used to sign the request.

Request Parameters

For information about the parameters that are common to all actions, See "Common Parameters" on page 423.

- `UserName`

  The name of the IAM user.

  This parameter is optional. If it is not included, it defaults to the user making the request.

  Type: String

  Length Constraints: Minimum length of 1. Maximum length of 64.

  Pattern: `[\w+=,.@-]+`

  Required: No

Response Elements

The following elements are returned by server.

- `AccessKeyMetadata.member.N`

  A list of objects that contains metadata about the access keys.

  Type: Array of `AccessKeyMetadata` objects See "Data Types" on page 444.

- `IsTruncated`

  A flag that indicates whether there are more items to return.

  Type: Boolean

Errors

For information about the errors that are common to all actions, See "Common Error Codes" on page 425.

- `NoSuchEntity`

  The request was rejected because it referenced a resource entity that does not exist. The error message describes the resource.

  HTTP Status Code: 404

- `ServiceFailure`

  The request processing has failed because of an unknown error, exception, or failure.

  HTTP Status Code: 500

Examples

Sample Request:

```
https://msdps3.veritas.com:8443/?Action=ListAccessKeys
&UserName=User1
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<ListAccessKeysResponse
  xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
  <ResponseMetadata>
    <RequestId>1648432612600646944</RequestId>
  </ResponseMetadata>
  <ListAccessKeysResult>
    <AccessKeyMetadata>
      <member>
        <AccessKeyId>2PPM4XHAKMG5JHZIUPEUG</AccessKeyId>
        <CreateDate>2022-03-28T01:43:46Z</CreateDate>
        <Status>Active</Status>
        <UserName>User1</UserName>
      </member>
      <member>
        <AccessKeyId>GAATH0QN9N5W8TBQPSKPJ</AccessKeyId>
        <CreateDate>2022-03-28T01:53:02Z</CreateDate>
        <Status>Active</Status>
        <UserName>User1</UserName>
      </member>
    </AccessKeyMetadata>
    <IsTruncated>false</IsTruncated>
```

```
  </ListAccessKeysResult>
</ListAccessKeysResponse>
```

DeleteAccessKey

Deletes the access key pair that is associated with the specified IAM user.

If you do not specify a user name, IAM determines the user name implicitly based on the MSDP S3 access key ID signing the request.

Request Parameters

For information about the parameters that are common to all actions, See "Common Parameters" on page 423.

- `AccessKeyId`
  The access key ID for the access key ID and secret access key you want to delete.
  Type: String
  Length Constraints: Minimum length of 16. Maximum length of 128.
  Pattern: `[\w]+`
  Required: Yes

- `UserName`
  The name of the IAM user.
  This parameter is optional. If it is not included, it defaults to the user making the request.
  Type: String
  Length Constraints: Minimum length of 1. Maximum length of 64.
  Pattern: `[\w+=,.@-]+`
  Required: No

Errors

For information about the errors that are common to all actions, See "Common Error Codes" on page 425.

- `NoSuchEntity`
  The request was rejected because it referenced a resource entity that does not exist. The error message describes the resource.
  HTTP Status Code: 404

- `ServiceFailure`
  The request processing has failed because of an unknown error, exception, or failure.
  HTTP Status Code: 500

Examples

Sample Request:

```
https://msdps3.veritas.com:8443/?Action=DeleteAccessKey
&AccessKeyId=GAATH0QN9N5W8TBQPSKPJ
&UserName=User1
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<DeleteAccessKeyResponse
  xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
  <ResponseMetadata>
    <RequestId>1648451304149485569</RequestId>
  </ResponseMetadata>
</DeleteAccessKeyResponse>
```

UpdateAccessKey

Changes the status of the specified access key from Active to Inactive, or vice versa. This operation can be used to disable a user's key as part of a key rotation workflow.

If the `UserName` is not specified, the user name is determined implicitly based on the MSDP S3 access key ID used to sign the request.

Request Parameters

For information about the parameters that are common to all actions, See "Common Parameters" on page 423.

- `AccessKeyId`

  The access key ID for the access key that you want to update.
  Type: String
  Length Constraints: Minimum length of 16. Maximum length of 128.
  Pattern: `[\w]+`
  Required: Yes

- `Status`

  The status you want to assign to the secret access key. Active means that the key can be used for programmatic calls to MSDP S3 server, while Inactive means that the key cannot be used.
  Type: String
  Valid Values: Active/Inactive
  Required: Yes

- `UserName`

  The name of the IAM user.

  This parameter is optional. If it is not included, it defaults to the user making the request.

  Type: String

  Length Constraints: Minimum length of 1. Maximum length of 64.

  Pattern: `[\w+=,.@-]+`

  Required: No

Errors

For information about the errors that are common to all actions, See "Common Error Codes" on page 425.

- `NoSuchEntity`

  The request was rejected because it referenced a resource entity that does not exist. The error message describes the resource.

  HTTP Status Code: 404

- `ServiceFailure`

  The request processing has failed because of an unknown error, exception, or, failure.

  HTTP Status Code: 500

Examples

Sample Request:

```
https://msdps3.veritas.com:8443/?Action=UpdateAccessKey
&AccessKeyId=GAATH0QN9N5W8TBQPSKPJ
&Status=Inactive
&UserName=User1
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<UpdateAccessKeyResponse
  xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
<ResponseMetadata>
    <RequestId>1648451481105563455</RequestId>
  </ResponseMetadata>
</UpdateAccessKeyResponse>
```

PutUserPolicy

Adds or updates an inline policy document that is embedded in the specified IAM user.

Request Parameters

For information about the parameters that are common to all actions, See "Common Parameters" on page 423.

- `PolicyDocument`
  The policy document.
  You must provide policies in JSON format in IAM.
  Type: String
  Required: Yes
- `PolicyName`
  The name of the policy document.
  Type: String
  Length Constraints: Minimum length of 1. Maximum length of 128.
  Pattern: `[\w+=,.@-]+`
  Required: Yes
- `UserName`
  The name of the user to associate the policy with.
  Type: String
  Length Constraints: Minimum length of 1. Maximum length of 64.
  Pattern: `[\w+=,.@-]+`
  Required: Yes

Errors

For information about the errors that are common to all actions, See "Common Error Codes" on page 425.

- `MalformedPolicyDocument`
  The request was rejected because the policy document was malformed.
  HTTP Status Code: 400
- `NoSuchEntity`
  The request was rejected because it referenced a resource entity that does not exist. The error message describes the resource.
  HTTP Status Code: 404
- `ServiceFailure`
  The request processing has failed because of an unknown error, exception, or failure.

HTTP Status Code: 500

Examples

Sample Request:

```
https://msdps3.veritas.com:8443/?Action=PutUserPolicy
&UserName=User1
&PolicyName=ExamplePolicy
&PolicyDocument={"Version":"2012-10-17","Statement":[{"Effect":"Allow",
"Action":["s3:*"],"Resource":["arn:aws:s3:::bkt3/*"]}]}
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<PutUserPolicyResponse
  xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
  <ResponseMetadata>
    <RequestId>1648451612346994599</RequestId>
  </ResponseMetadata>
</PutUserPolicyResponse>
```

GetUserPolicy

Retrieves the specified inline policy document that is embedded in the specified IAM user.

Request Parameters

For information about the parameters that are common to all actions, See "Common Parameters" on page 423.

- `PolicyName`
  The name of the policy document to get.
  Type: String
  Length Constraints: Minimum length of 1. Maximum length of 128.
  Pattern: `[\w+=,.@-]+`
  Required: Yes

- `UserName`
  The name of the user who the policy is associated with.
  Type: String
  Length Constraints: Minimum length of 1. Maximum length of 64.
  Pattern: `[\w+=,.@-]+`
  Required: Yes

Response Elements

The following elements are returned by server.

- `PolicyDocument`
  The policy document.
  IAM stores policies in JSON format.
  Type: String
- `PolicyName`
  The name of the policy.
  Type: String
  Length Constraints: Minimum length of 1. Maximum length of 128.
  Pattern: `[\w+=,.@-]+`
- `UserName`
  The user the policy is associated with.
  Type: String
  Length Constraints: Minimum length of 1. Maximum length of 128.
  Pattern: `[\w+=,.@-]+`

Errors

For information about the errors that are common to all actions, See "Common Error Codes" on page 425.

- `NoSuchEntity`
  The request was rejected because it referenced a resource entity that does not exist. The error message describes the resource.
  HTTP Status Code: 404
- `ServiceFailure`
  The request processing has failed because of an unknown error, exception, or failure.
  HTTP Status Code: 500

Examples

Sample Request:

```
https://msdps3.veritas.com:8443/?Action=GetUserPolicy
&UserName=User1
&PolicyName=ExamplePolicy
&Version=2010-05-08
&AUTHPARAMS
```

Sample Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<GetUserPolicyResponse
  xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
  <ResponseMetadata>
    <RequestId>1648441417449582212</RequestId>
  </ResponseMetadata>
  <GetUserPolicyResult>
    <UserName>User1</UserName>
    <PolicyName>ExamplePolicy</PolicyName>
    <PolicyDocument>{"Version":"2012-10-17","Statement":[{"Effect":"Allow",
    "Action":["s3:*"],"Resource":["arn:aws:s3:::bkt3/*"]}]}</PolicyDocument>
  </GetUserPolicyResult>
</GetUserPolicyResponse>
```

ListUserPolicies

Lists the names of the inline policies that are embedded in the specified IAM user.

Request Parameters

For information about the parameters that are common to all actions, See "Common Parameters" on page 423.

- `UserName`

  The name of the user to list policies for.

  Type: String

  Length Constraints: Minimum length of 1. Maximum length of 64.

  Pattern: `[\w+=,.@-]+`

  Required: Yes

Response Elements

The following element is returned by server.

- `PolicyNames.member.N`

  A list of policy names.

  Type: Array of strings

  Length Constraints: Minimum length of 1. Maximum length of 128.

  Pattern: `[\w+=,.@-]+`

- `IsTruncated`

  A flag that indicates whether there are more items to return.

  Type: Boolean

Errors

For information about the errors that are common to all actions, See "Common Error Codes" on page 425.

- `NoSuchEntity`

  The request was rejected because it referenced a resource entity that does not exist. The error message describes the resource.

  HTTP Status Code: 404

- `ServiceFailure`

  The request processing has failed because of an unknown error, exception, or failure.

  HTTP Status Code: 500

Examples

Sample Request:

```
https://msdps3.veritas.com:8443/?Action=ListUserPolicies
&UserName=User1
&AUTHPARAMS
```

Sample Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<ListUserPoliciesResponse
  xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
  <ResponseMetadata>
    <RequestId>1648441729868934088</RequestId>
  </ResponseMetadata>
  <ListUserPoliciesResult>
    <PolicyNames>
      <member>ExamplePolicy</member>
    </PolicyNames>
    <IsTruncated>false</IsTruncated>
  </ListUserPoliciesResult>
</ListUserPoliciesResponse>
```

DeleteUserPolicy

Deletes the specified inline policy that is embedded in the specified IAM user.

Request Parameters

For information about the parameters that are common to all actions, See "Common Parameters" on page 423.

- `PolicyName`

  The name identifying the policy document to delete.

Type: String
Length Constraints: Minimum length of 1. Maximum length of 128.
Pattern: `[\w+=,.@-]+`
Required: Yes

- `UserName`
  The name identifying the user that the policy is embedded in.
  Type: String
  Length Constraints: Minimum length of 1. Maximum length of 64.
  Pattern: `[\w+=,.@-]+`
  Required: Yes

Errors

For information about the errors that are common to all actions, See "Common Error Codes" on page 425.

- `NoSuchEntity`
  The request was rejected because it referenced a resource entity that does not exist. The error message describes the resource.
  HTTP Status Code: 404
- `ServiceFailure`
  The request processing has failed because of an unknown error, exception, or failure.
  HTTP Status Code: 500

Examples

Sample Request:

```
https://msdps3.veritas.com:8443/?Action=DeleteUserPolicy
&PolicyName=ExamplePolicy
&UserName=User1
&AUTHPARAMS
```

Sample Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<DeleteUserPolicyResponse
  xmlns="https://iam.amazonaws.com/doc/2010-05-08/">
  <ResponseMetadata>
    <RequestId>1648451943468940588</RequestId>
  </ResponseMetadata>
</DeleteUserPolicyResponse>
```

Data Types

Table 8-3 Data types

Data type	Description
`User`	Contains the information about an IAM user entity. ■ `UserName` The friendly name identifying the user. Type: String Length Constraints: Minimum length of 1. Maximum length of 64. Pattern: `[\w+=,.@-]+` Required: Yes ■ `CreateDate` The date and time, in ISO 8601 date-time format, when the user was created. Type: Timestamp Required: Yes

Table 8-3 Data types *(continued)*

Data type	Description
`AccessKey`	Contains the information about an MSDP S3 access key. ■ `AccessKeyId` The ID for this access key. Type: String Length Constraints: Minimum length of 16. Maximum length of 128. Pattern: `[\w]+` Required: Yes ■ `CreateDate` The date when the access key was created. Type: Timestamp Required: No ■ `SecretAccessKey` The secret key that is used to sign requests. Type: String Required: Yes ■ `Status` The status of the access key. Active means that the key is valid for API calls, while Inactive means it is not. Type: String Valid Values: Active | Inactive Required: Yes ■ `UserName` The name of the IAM user that the access key is associated with. Type: String Length Constraints: Minimum length of 1. Maximum length of 64. Pattern: `[\w+=,.@-]+` Required: Yes

Table 8-3 Data types *(continued)*

Data type	Description
`AccessKeyMetadata`	Contains the information about an MSDP S3 access key, without its secret key. ■ `AccessKeyId` The ID for this access key. Type: String Length Constraints: Minimum length of 16. Maximum length of 128. Pattern: `[\w]+` Required: No ■ `CreateDate` The date when the access key was created. Type: Timestamp Required: No ■ `Status` The status of the access key. `Active` means that the key is valid for API calls, while `Inactive` means that it is not. Type: String Valid Values: Active | Inactive Required: No ■ `UserName` The name of the IAM user that the access key is associated with. Type: String Length Constraints: Minimum length of 1. Maximum length of 64. Pattern: `[\w+=,.@-]+` Required: No

IAM policy document syntax

A policy document is a JSON format document that contains Version and Statement objects. For example,

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
```

```
        "s3:Get*",
        "s3:List*"
      ],
      "Resource": [
        "arn:aws:s3:::*"
      ]
    }
  ]
}
```

Supported Version in policy document:

Only "2012-10-17" is supported.

Supported Action:

Table 8-4 Supported Action

Action	Description	Permissive APIs
`s3:*`	Any S3 and IAM operations. This is an administrator permission.	All S3 and IAM APIs. **Note: CreateBucket** API requires this permission. The permission `s3:BypassGovernanceRetention` is only applied to the action `s3:*`.
`s3:Put*`	S3 write operations.	`UploadPart` `CompleteMultipartUpload` `CreateMultipartUpload` `AbortMultipartUpload` `PutObject` `DeleteObject` `DeleteObjects` `PutBucketVersioning` `DeleteBucket` `CopyObject` `PutObjectLockConfiguration` `PutObjectRetention`

Table 8-4 Supported Action *(continued)*

Action	Description	Permissive APIs
`s3:Get*`	S3 read operations.	`HeadObject` `GetObject` `GetBucketVersioning` `GetBucketLocation` `GetBucketEncryption` `HeadBucket` `CopyObject` `GetObjectLockConfiguration` `GetObjectRetention`
`s3:List*`	S3 list operations.	`ListBuckets` `ListObjects` `ListObjectsV2` `ListObjectVersions` `ListMultipartUploads`

Supported Effect:

Only "Allow" effect is supported.

Note: root user has embedded administrator permission, so you cannot attach a policy to root user.

Supported Resource patterns:

Table 8-5 Supported Resource patterns

Resource pattern	Description
`arn:aws:s3:::*`	All S3 resources. **Note:** If this resource pattern is used with action `s3:*`, it means that the user has all permissions for all S3 resources, which are same as a root user. The permission `s3:BypassGovernanceRetention` is only applied to the action `s3:*`.
`arn:aws:s3:::<BUCKET_NAME>/*`	All objects within <BUCKET_NAME>. And the bucket itself. The permission `s3:BypassGovernanceRetention` is not applied to the current resource.

S3 Object Lock In Flex WORM

S3 Object Lock lets you store the objects using a write once read many (WORM) model. Currently, the feature works only in Flex WORM and the legal hold APIs are not supported.

We recommend that you store all the data into the object-lock-enabled buckets. The setting for S3 Object Lock in Flex WORM and MSDP in Flex WORM are same by default. However, the format of object lock retention is different. For example, "Days" or "Years" in S3 bucket object lock retention is different from the MSDP WORM storage server, which uses "Hours" to "Years". We recommend that you enlarge the interval range to minimize the effects on the backup retention.

If the MSDP WORM settings are updated, you must restart **s3srv** and the **MSDP** service. After the WORM settings are updated, the retention settings of the existing objects stay unchanged and only the newly created objects are affected.

Note: The Governance mode in Flex WORM S3 object lock is Enterprise mode in the MSDP LSU on Flex WORM. The Compliance mode in Flex WORM S3 object lock is Compliance mode in the MSDP LSU on Flex WORM.

You can use the following S3 APIs for Object Lock in Flex WORM:

- Create Bucket
  See “CreateBucket ” on page 451.

- Put Object
  See “PutObject” on page 492.
- Copy Object
  See “Copy Object” on page 493.
- Get Object
  See “GetObject” on page 487.
- Head Object
  See “HeadObject” on page 490.
- Delete Object
  See “DeleteObject” on page 483.
- Delete Objects
  See “DeleteObjects” on page 485.
- Create Multipart Upload
  See “CreateMultipartUpload” on page 482.
- Put Object Retention (Flex WORM only)
  See “Put Object Retention (Flex WORM only)” on page 499.
- Get Object Retention (Flex WORM only)
  See “Get Object Retention (Flex WORM only)” on page 500.
- Put Object Lock Configuration (Flex WORM only)
  See “Put Object Lock Configuration (Flex WORM only)” on page 474.
- GET Object Lock Configuration (Flex WORM only)
  See “Get Object Lock Configuration (Flex WORM only)” on page 476.

S3 APIs for S3 interface for MSDP

Common Error Response

In case of error following xml response is returned to REST client.

- Response content type is "application/xml"
- `RequestId` is unique ID generated per request.
- Response content is in the following XML format:

```
<?xml version="1.0" encoding="UTF-8"?>
<Error>
    <Code>InvalidKeyMarker</Code>
    <Message>The key marker is invalid. It should start with prefix.
    </Message>
```

```
<Resource>/azure-versioned/</Resource>
<RequestId>16533774727514537 58</RequestId>
```

S3 APIs on Buckets

S3 APIs on buckets perform the following data arrangement functions:

- Create a bucket.
- Delete a bucket.
- Check the bucket status.
- List the buckets.

CreateBucket

Creates a new bucket. The bucket name is global unique for different LSU. Not every string is an acceptable bucket name. For information about bucket naming restrictions, see Bucket naming rules. You must specify `Region(=lsu name)` in the request body. You are not permitted to create buckets using anonymous requests.

Request Syntax

```
PUT /bucket HTTP/1.1
Host: msdps3.server:8443
<?xml version="1.0" encoding="UTF-8"?>
<CreateBucketConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
   <LocationConstraint>string</LocationConstraint>
</CreateBucketConfiguration>
```

Request Parameters

- `Bucket`
  Name of the bucket to be created.
  Required: Yes
  Type: String
- `x-amz-bucket-object-lock-enabled` (Flex WORM only)
  Specifies if S3 Object Lock is enabled for the new bucket.

Note: If `ObjectLockEnabledForBucket` is set to true in the `CreateBucket` request, `s3:PutObjectLockConfiguration` permission is required and the bucket versioning is enabled automatically. `s3:PutBucketVersioning` permission is not required.

Request Body

- `CreateBucketConfiguration`
  Root level tag for the `CreateBucketConfiguration` parameters.
  Required: Yes
- `LocationConstraint`
  Specifies the Region where the bucket will be created.

  Note: The regions in **S3Srv** are the LSU names. If you don't specify a region, the bucket is created in the region **PureDiskVolume** (Local LSU).

  Type: String
  Valid Values: PureDiskVolume, CLOUD_LSU_NAME
  Required: No

Response Syntax

```
HTTP/1.1 200
```

Possible Error Response

- `Success`
  HTTP status code 200.
- `InvalidArgument`
  Invalid Argument.
  HTTP status code 400.
- `InvalidBucketName`
  The specified bucket is not valid.
  HTTP status code 400.
- `AccessDenied`
  Access Denied.
  HTTP status code 403.
- `BucketAlreadyExists`
  The requested bucket name is not available. The bucket namespace is shared by all users of the system. Please select a different name and try again.
  HTTP status code 409.
- `InternalError`
  Request failed because of an internal server error.
  HTTP status code 500.

DeleteBucket

Deletes the bucket. All objects including all object versions and delete markers in the bucket must be deleted before the bucket itself can be deleted.

Request Syntax

```
DELETE /bucket HTTP/1.1
Host: msdps3.server:8443
```

Request Parameters

- `Bucket`
  Name of the bucket to be deleted.
  Required: Yes
  Type: String

Response Syntax

```
HTTP/1.1 204
```

Possible Error Response

- `Success`
  HTTP status code 204.
- `AccessDenied`
  Access Denied.
  HTTP status code 403.
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.
- `BucketNotEmpty`
  The bucket you tried to delete is not empty.
  HTTP status code 409.
- `InternalError`
  Request failed because of an internal server error.
  HTTP status code 500.

GetBucketEncryption

Returns the default encryption configuration for a bucket.

Request Syntax

```
GET /bucket?encryption HTTP/1.1
Host: msdps3.server:8443
```

Request Parameters

- `Bucket`
  Name of the bucket.
  Required: Yes
  Type: String

Response Syntax

```
HTTP/1.1 200 <?xml version="1.0" encoding="UTF-8"?>
<ServerSideEncryptionConfiguration>
    <Rule>
        <ApplyServerSideEncryptionByDefault>
            <SSEAlgorithm>string</SSEAlgorithm>
        </ApplyServerSideEncryptionByDefault>
    </Rule>
</ServerSideEncryptionConfiguration>
```

Response Body

- `ServerSideEncryptionConfiguration`
  Root level tag for the `ServerSideEncryptionConfiguration` parameters.
  Required: Yes
  - Rule
    Container for information about a particular server-side encryption configuration rule.
    - `ApplyServerSideEncryptionByDefault`
      Specifies the default server-side encryption to apply to objects in the bucket.
      - `SSEAlgorithm`
        Server-side encryption algorithm to use for the default encryption.

Possible Error Response

- `Success`
  HTTP status code 200.
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.
- `InternalError`

Request failed because of an internal server error.
HTTP status code 500.

GetBucketLocation

Returns the bucket's region using `LocationConstraint` of that object. Bucket's region is MSDP LSU.

Request Syntax

```
GET /bucket?location HTTP/1.1
Host: msdps3.server:8443
```

Request Parameters

- `Bucket`
  Name of the bucket.
  Required: Yes
  Type: String

Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<LocationConstraint>
    <LocationConstraint>string</LocationConstraint>
</LocationConstraint>
```

Response Body

- `LocationConstraint`
  Root level tag for the `LocationConstraint` parameters.
  Required: Yes
- `LocationConstraint`
  `LocationConstraint` of that object is MSDP LSU.

Possible Error Response

- `Success`
  HTTP status code 200.
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.
- `InternalError`
  Request failed because of an internal server error.

HTTP status code 500.

GetBucketVersioning

Returns the versioning state of a bucket.

Request Syntax

```
GET /bucket?versioning HTTP/1.1
Host: msdps3.server:8443
```

Request Parameters

- `Bucket`
  Bucket name for which you want to get the versioning information.
  Required: Yes
  Type: String

Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<VersioningConfiguration>
    <Status>string</Status>
</VersioningConfiguration>
```

Response Body

- `VersioningConfiguration`
  Root level tag for the `VersioningConfiguration` parameters.
  Required: Yes
- `Status`
  Versioning status of bucket.
  Valid Values: Enabled

Possible Error Response

- `Success`
  HTTP status code 200.
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.
- `InternalError`
  Request failed because of an internal server error.
  HTTP status code 500.

HeadBucket

Determines if a bucket exists or not. The operation returns a 200 OK if the bucket exists and the user has permissions to access it.

Request Syntax

```
HEAD /bucket HTTP/1.1
Host: msdps3.server:8443
```

Request Parameters

- `Bucket`
  The name of the bucket.
  Required: Yes
  Type: String

Response Syntax

```
HTTP/1.1 200
```

Possible Error Response

- `Success`
  HTTP status code 200.
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.
- `InternalError`
  Request failed because of an internal server error.
  HTTP status code 500.

ListBuckets

Lists all the buckets.

Request Syntax

```
GET / HTTP/1.1
Host: msdps3.server:8443
```

Request Parameters

The request does not use any URI parameters.

Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ListAllMyBucketsResult>
    <Buckets>
        <Bucket>
            <CreationDate>timestamp</CreationDate>
            <Name>string</Name>
        </Bucket>
    </Buckets>
</ListAllMyBucketsResult>
```

Response Body

- `ListAllMyBucketsResult`
  Root level tag for all bucket results.
  Required: Yes
  - `Buckets`
    The list of buckets owned by the user that is authenticated for the request.
    - `Bucket`
      Information of the bucket.
      - `CreationDate`
        Bucket creation date and time.
      - `Name`
        Name of the bucket.

Possible Error Response

- `Success`
  HTTP status code 200.
- `AccessDenied`
  Request was rejected because user authentication failed.
  HTTP status code 403.
- `InternalError`
  Request failed because of an internal server error.
  HTTP status code 500.

ListMultipartUploads

Lists in-progress multipart uploads. An in-progress multipart upload is a multipart upload that is initiated using the Create Multipart Upload request but is not complete yet or aborted. This operation randomly returns a maximum of 10000 multipart

uploads in the response that is sorted by object key in ascending order. The operation does not support paging.

Request Syntax

```
GET /bucket?uploads&prefix=Prefix
Host: msdps3.server:8443
```

Request Parameters

- `Bucket`
  Name of the bucket on which the multipart upload was initiated.
  Required: Yes
  Type: String
- `prefix`
  Limits the response to uploads that begin with the specified prefix.
  Type: String

Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ListMultipartUploadsResult>
    <Bucket>string</Bucket>
    <KeyMarker>string</KeyMarker>
    <UploadIdMarker>string</UploadIdMarker>
    <NextKeyMarker>string</NextKeyMarker>
    <Prefix>string</Prefix>
    <NextUploadIdMarker>string</NextUploadIdMarker>
    <MaxUploads>integer</MaxUploads>
    <IsTruncated>boolean</IsTruncated>
    <Upload>
        <Initiated>timestamp</Initiated>
        <Key>string</Key>
        <StorageClass>string</StorageClass>
        <UploadId>string</UploadId>
    </Upload>
    ...
</ListMultipartUploadsResult>
```

Response Body

- `ListMultipartUploadsResult`
  Root level tag for the `ListMultipartUploadsResult` parameters.
  Required: Yes

- `Bucket`
  Name of the bucket on which the multipart upload was initiated.
- `IsTruncated`
  A flag indicating whether all the results satisfying the search criteria were returned by MSDP S3.
- `KeyMarker`
  S3 interface for MSDP expects the key-marker which was returned by server in last request. The value of "NextKeyMarker" of response should be used in request as key-marker.
- `MaxUploads`
  Limits the number of multipart uploads that are returned in the response.
- `NextKeyMarke`
  When the response is truncated, you can use this value as marker in subsequent request to get next set of objects.
- `NextUploadIdMarker`
  When the response is truncated, you can use this value as marker in subsequent request to get next set of objects.
- `UploadIdMarker`
  The value of `UploadIdMarker` passed in the request.
- `Prefix`
  Limits the response to keys that begin with the specified prefix.
- `Upload`
  Information that is related to a particular multipart upload. Response can contain zero or multiple uploads.
  - `Initiated`
    The time and date when the multipart upload was initiated.
    Type: Timestamp
  - `Key`
    Object name for which multipart upload was initiated.
  - `StorageClass`
    Storage class of the uploaded part.
  - `UploadId`
    Upload ID that identifies the multipart upload.

Possible Error Response

- `Success`

HTTP status code 200.

- `InvalidArgument`
  Invalid Argument. HTTP status code 400.
- `AccessDenied`
  Request was rejected because user authentication failed.
  HTTP status code 403.
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.
- `InternalError`
  Request failed because of an internal server error.
  HTTP status code 500.

ListObjects

Returns a list of all the objects in a bucket. You can use the request parameters as selection criteria to return a subset of the objects in a bucket. The API returns objects with the latest version when the versioning is enabled on the bucket. A 200 OK response can contain valid or invalid XML. Ensure that you design your application to parse the contents of the response and handle it appropriately.

Request Syntax

```
GET /bucket?delimiter=Delimiter&marker=Marker&max-keys
=Maxkeys&prefix=Prefix HTTP/1.1
Host: msdps3.server:8443
```

Request Parameters

- `Bucket`
  Name of the bucket that contains the objects.
  Required: Yes
  Type: String
- `delimiter`
  A delimiter is a character used to group keys. It rolls up the keys that contain the same character between the prefix and the first occurrence of the delimiter into a single result element in the `CommonPrefixes` collection. These rolled-up keys are not returned elsewhere in the response. Each rolled-up result counts as only one return against the `MaxKeys` value. MSDP S3 supports only "/" string as delimiter.
  Type: String

- `marker`

  The marker is the point where S3 interface for MSDP should begin listing objects. S3 interface for MSDP expects the marker which was returned by server in last request. The value of `NextMarker` of response should be used in request as marker.

  Type: String

- `max-keys`

  Limits the number of keys that are returned in the response. By default, the action returns up to 1,000 key names.

  Type: Integer

- `prefix`

  Limits the response to keys that begin with the specified prefix.

  Type: String

Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ListBucketResult>
    <IsTruncated>boolean</IsTruncated>
    <Marker>string</Marker>
    <NextMarker>string</NextMarker>
    <Contents>
        <ETag>string</ETag>
        <Key>string</Key>
        <LastModified>timestamp</LastModified>
        <Size>integer</Size>
        <StorageClass>string</StorageClass>
    </Contents>
    ...
    <Name>string</Name>
    <Prefix>string</Prefix>
    <Delimiter>string</Delimiter>
    <MaxKeys>integer</MaxKeys>
    <CommonPrefixes>
        <Prefix>string</Prefix>
    </CommonPrefixes>
    ...
</ListBucketResult>
```

Response Body

- `ListBucketResult`

Root level tag for the `ListBucketResult` parameters.

Required: Yes

- `CommonPrefixes`

  When determining the number of returns, all the keys (up to 1,000) rolled into a common prefix count as one. `CommonPrefixes` contains all keys between Prefix and the next occurrence of the string that is specified by the delimiter.

- `Delimiter`

  Delimiter value that is passed in request.

- `IsTruncated`

  A flag indicating whether all the results satisfying the search criteria were returned by MSDP S3.

- `Marker`

  Indicates where listing begins in the bucket. Marker is included in response only if it is passed in request.

- `MaxKeys`

  The maximum number of objects that can be returned in the response body.

- `Name`

  The name of the bucket

- `NextMarker`

  When the response is truncated, you can use this value as the marker in the subsequent request to get the next set of objects.

- `Prefix`

  Limits the response to keys that begin with the specified prefix.

- `Contents`

  Metadata about each object that is returned.

  - `ETag`

    SHA256 digest of the object.

  - `Key`

    Object name.

  - `LastModified`

    Last modification date and time of the object.

  - `Size`

    Size of the object.

  - `StorageClass`

Storage class of the object.

For versioned bucket, it is recommended that you use List Object Versions API to get information about all objects. If using "list objects" in versioned bucket when results are truncated, key count in the results may be less than Max keys and you can make a follow-up paginated request.

When using **list objects** APIs on a versioned bucket, if all of objects under the specified prefix are delete markers, the specified prefix is displayed as a `CommonPrefixes` element.

Possible Error Response

- `Success`
  HTTP status code 200.
- `InvalidArgument(Marker is invalid.)`
  Invalid Argument. HTTP status code 400.
- `InvalidArgument(maxKeys is invalid)`
  Invalid Argument.
  HTTP status code 400.
- `S3srvExtInvalidPrefix`
  Prefixes cannot start with a slash.
  HTTP status code 400.
- `S3srvExtInvalidDelimiter`
  Only support the slash as a delimiter.
  HTTP status code 400
- `AccessDenied`
  Access Denied.
  HTTP status code 403.
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.
- `InternalError`
  Request failed because of an internal server error.
  HTTP status code 500.

ListObjectsV2

Returns a list of all the objects in a bucket. You can use the request parameters as a selection criteria to return a subset of the objects in a bucket. The API returns objects with the latest version when the versioning is enabled on the bucket. A 200

OK response can contain valid or invalid XML. Make sure to design your application to parse the contents of the response and handle it appropriately.

Request Syntax

```
GGET /bucket?list-type=2&continuation-token=Continuation
Token&delimiter=Delimiter&max-keys=MaxKeys&prefix=Prefix HTTP/1.1
Host: msdps3.server:8443
```

Request Parameters

- `Bucket`
  Name of the bucket that contains the objects.
  Required: Yes
  Type: String
- `continuation-token`
  Continuation-token is the point from where you want S3 interface for MSDP to start listing objects. S3 interface for MSDP expects the continuation-token that is returned by the server in the last request. The value of `NextContinuationToken` of response should be used in request as `ContinuationToken`. The token can only be used once and valid for two minutes by default.
  Type: String
- `delimiter`
  A delimiter is a character used to group keys. It rolls up the keys that contain the same character between the prefix and the first occurrence of the delimiter into a single result element in the `CommonPrefixes` collection. These rolled-up keys are not returned elsewhere in the response. Each rolled-up result counts as only one return against the `MaxKeys` value. MSDP S3 supports only "/" string as delimiter.
  Type: String
- `max-keys`
  Limits the number of keys that are returned in the response. By default, the action returns up to 1,000 key names.
  Type: Integer
- `prefix`
  Limits the response to keys that begin with the specified prefix.
  Type: String

Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
```

```
<ListBucketResult>
    <IsTruncated>boolean</IsTruncated>
    <Contents>
        <ETag>string</ETag>
        <Key>string</Key>
        <LastModified>timestamp</LastModified>
        <Size>integer</Size>
        <StorageClass>string</StorageClass>
    </Contents>
    ...
    <Name>string</Name>
    <Prefix>string</Prefix>
    <Delimiter>string</Delimiter>
    <MaxKeys>integer</MaxKeys>
    <CommonPrefixes>
        <Prefix>string</Prefix>
    </CommonPrefixes>
    ...
    <KeyCount>integer</KeyCount>
    <ContinuationToken>string</ContinuationToken>
    <NextContinuationToken>string</NextContinuationToken>
</ListBucketResult>
```

Response Body

- `ListBucketResult`

  Root level tag for the `ListBucketResult` parameters.

  Required: Yes

  - `CommonPrefixes`

    When determining the number of returns, all the keys (up to 1,000) rolled into a common prefix count as one. `CommonPrefixes` contains all keys between Prefix and the next occurrence of the string that is specified by the delimiter.

  - `Contents`

    Metadata about each object that is returned.

    - `ETag`

      SHA256 digest of the object.

    - `Key`

      Object name.

    - `LastModified`

      Last modification date and time of the object.

- `Size`
  Size of the object.
- `StorageClass`
  Storage class of the object.

- `Delimiter`
  Delimiter value that is passed in request.
- `IsTruncated`
  A flag indicating whether all the results satisfying the search criteria were returned by MSDP S3.
- `ContinuationToken`
  The `ContinuationToken` is the point from where you want S3 interface for MSDP to start listing objects. S3 interface for MSDP expects `ContinuationToken` that is returned by server in last request. The value of `NextContinuationToken` of response should be used in request as `ContinuationToken`.
- `KeyCount`
  The number of objects that are returned in the response body.
- `MaxKeys`
  The maximum number of objects that can be returned in the response body.
- `Name`
  The name of the bucket
- `NextContinuationToken`
  When the response is truncated, you can use this value as `ContinuationToken` in subsequent request to get next set of objects.
- `Prefix`
  Limits the response to keys that begin with the specified prefix.

For versioned bucket, it is recommended that you use **List Object Versions** API to get all objects information. If using "list objects" in versioned bucket when results are truncated, key count in the results may be less than Max keys and you can make a follow-up paginated request.

Recommend less than 1000 `CommonPrefixes` elements under a specified prefix separated by the slash (/) delimiter character. If more than 10000 `CommonPrefixes` elements under a specified prefix exist, list objects with the prefix and the delimiter parameters in the request returns only 10000 elements. You can use list objects without delimiter if you want to list all elements under a specified prefix.

When using **list objects** APIs on a versioned bucket, if all of objects under the specified prefix are delete markers, the specified prefix is shown as a `CommonPrefixes` element.

Possible Error Response

- `Success`
  HTTP status code 200.
- `InvalidArgument(continuation-token is invalid)`
  Invalid Argument.
  HTTP status code 400.
- `InvalidArgument(max-keys is invalid)`
  Invalid Argument.
  HTTP status code 400.
- `S3srvExtInvalidPrefix`
  Prefixes cannot start with a slash.
  HTTP status code 400.
- `S3srvExtInvalidDelimiter`
  Only support the slash as a delimiter.
  HTTP status code 400
- `AccessDenied`
  Access Denied.
  HTTP status code 403.
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.
- `InternalError`
  Request failed because of an internal server error.
  HTTP status code 500.

ListObjectVersions

Returns metadata about all versions of the objects in a bucket. You can also use request parameters as selection criteria to return metadata about a subset of all the object versions. S3 interface for MSDP recommends using this API with 1000 max keys and object name as a prefix to list all object versions in one request.

Request Syntax

```
GET /bucket/?versions&delimiter=Delimiter&key-marker=
KeyMarker&max-keys=MaxKeys&prefix=Prefix HTTP/1.1
Host: msdps3.server:8443
```

Or

```
GET /bucket/?versions&delimiter=Delimiter&max-keys=
MaxKeys&prefix=Prefix&version-id-marker=VersionIdMarker HTTP/1.1
Host: msdps3.server:8443
```

Request Parameters

- `Bucket`
  Name of the bucket that contains the objects.
  Required: Yes
  Type: String
- `key-marker`
  The value of `NextKeyMarker` of response should be used in request as marker. The marker can only be used once and valid for two minutes by default. This parameter can only be used with `version-id-marker`.
  Type: String
- `delimiter`
  A delimiter is a character used to group keys. It rolls up the keys that contain the same character between the prefix and the first occurrence of the delimiter into a single result element in the `CommonPrefixes` collection. These rolled-up keys are not returned elsewhere in the response. Each rolled-up result counts as only one return against the `MaxKeys` value. MSDP S3 supports only "/" string as delimiter.
  Type: String
- `max-keys`
  Limits the number of keys returned in the response. By default, the action returns up to 1,000 key names.
  Type: Integer
- `prefix`
  Limits the response to keys that begin with the specified prefix.
  Type: String
- `version-id-marker`
  The value of `NextVerionIDMarker` of response should be used in request as `VersionIdMarker`. The marker can only be used once and valid for two minutes by default. This parameter can only be used with `key-marker`.
  Type: String

Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ListVersionsResult>>
    <IsTruncated>boolean</IsTruncated>
    <KeyMarker>string</KeyMarker>
    <VersionIdMarker>string</VersionIdMarker>
    <NextKeyMarker>string</NextKeyMarker>
    <NextVersionIdMarker>string</NextVersionIdMarker>
    <Version>
        <ETag>string</ETag>
        <IsLatest>boolean</IsLatest>
        <Key>string</Key>
        <LastModified>timestamp</LastModified>
        <Size>integer</Size>
        <StorageClass>string</StorageClass>
        <VersionId>string</VersionId>
    </Version>
    ...
    <DeleteMarker>
        <IsLatest>boolean</IsLatest>
        <Key>string</Key>
        <LastModified>timestamp</LastModified>
        <VersionId>string</VersionId>
    </DeleteMarker>
    ...
    <Name>string</Name>
    <Prefix>string</Prefix>
    <Delimiter>string</Delimiter>
    <MaxKeys>integer</MaxKeys>
    <CommonPrefixes>
        <Prefix>string</Prefix>
    </CommonPrefixes>
    ...
</ListVersionsResult>>
```

Response Body

- `ListVersionsResult`

  Root level tag for the `ListVersionsResult` parameters.

  Required: Yes

  - `DeleteMarker`

Metadata about each delete marker. Response can have zero or more delete markers.

- Contents
  Metadata about each object that is returned.
  - `IsLatest`
    Specify if the object is latest.
    Type: Boolean
  - `Key`
    Delete marker name.
  - `LastModified`
    Last modification date and time of the delete marker.
    Type: Timestamp
  - `VersionId`
    Specify version ID of the delete marker.
- `Delimiter`
  Delimiter value that is passed in the request.
- `IsTruncated`
  A flag indicating whether all the results satisfying the search criteria were returned by MSDP S3.
- `KeyMarker`
  The value of `NextKeyMarker` of response should be used in request as `KeyMarker`.
- `MaxKeys`
  The maximum number of objects that can be returned in the response body.
- `Name`
  The name of the bucket.
- `NextKeyMarker`
  When the response is truncated, you can use this value as `KeyMarker` in subsequent request to get next set of objects.
- `NextVersionIdMarker`
  When the response is truncated, you can use this value as `VersionIdMarker` in subsequent request to get next set of objects.
- `Prefix`
  Limits the response to keys that begin with the specified prefix.
- `VersionIdMarker`

The value of `NextVersionIdMarker` of response should be used in request as `VersionIdMarker`.

- `Version`
  Metadata about object version.
  - `ETag`
    SHA256 digest of the object.
  - `IsLatest`
    Specify if the object is latest.
    Type: Boolean
  - `Key`
    Object name.
  - `LastModified`
    Last modification date and time of the object.
  - `DELETE /bucket/Key+?uploadId=UploadId HTTP/1.1`
    Size of the object.
  - `StorageClass`
    Storage class of the object.
  - `VersionId`
    Specify version ID of the object.

Recommend less than 1000 `CommonPrefixes` elements under a specified prefix that is separated by the slash (/) delimiter character. If more than 10000 `CommonPrefixes` elements under a specified prefix exist, list objects with the prefix and the delimiter parameters in the request returns only 10000 elements. You can use list objects without delimiter if you want to list all elements under a specified prefix.

Possible Error Response

- `Success`
  HTTP status code 200.
- `InvalidArgument(continuation token is invalid)`
  Invalid Argument. HTTP status code 400.
- `InvalidArgument(maxKeys is invalid)`
  Invalid Argument.
  HTTP status code 400.
- `S3srvExtInvalidPrefix`
  Prefixes cannot start with a slash.

HTTP status code 400.

- `S3srvExtInvalidDelimiter`
  Only support the slash as a delimiter.
  HTTP status code 400
- `S3srvExtInvalidKeyMarker`
  The key marker is invalid.
  HTTP status code 400
- `AccessDenied`
  Access Denied.
  HTTP status code 403.
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.
- `InternalError`
  Request failed because of an internal server error.
  HTTP status code 500.

PutBucketVersioning

Sets the versioning state of an existing bucket. You can set the versioning state with the value `Enabled`, which enables versioning for the objects in the bucket.

If the versioning state has never been set on a bucket, the bucket has no versioning state. When you enabled versioning on the bucket, the bucket is in the versioning state and cannot be set back to non-versioning state.

Request Syntax

```
PUT /bucket/?versioning HTTP/1.1
Host: msdps3.server:8443
<?xml version="1.0" encoding="UTF-8"?>
<VersioningConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
   <Status>string</Status>
</VersioningConfiguration>
```

Request Parameters

- `Bucket`
  Name of the bucket that contains the objects.
  Required: Yes
  Type: String

Request body

- `Status`
  The versioning state of the bucket.
  Valid Values: Enabled
  Required: Yes
  Type: String

Response Syntax

```
HTTP/1.1 200
```

Possible Error Response

- `Success`
  HTTP status code 200.
- `AccessDenied`
  Access Denied.
  HTTP status code 403.
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.
- `InternalError`
  Request failed because of an internal server error.
  HTTP status code 500.

Put Object Lock Configuration (Flex WORM only)

Places an Object Lock configuration on the specified bucket. The rule specified in the Object Lock configuration is applied by default to every new object placed in the specified bucket.

Request Syntax

```
PUT /{bucket}/?object-lock HTTP/1.1
Host: msdps3.server:8443

<?xml version="1.0" encoding="UTF-8"?>
<ObjectLockConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
   <ObjectLockEnabled>string</ObjectLockEnabled>
   <Rule>
      <DefaultRetention>
         <Days>integer</Days>
         <Mode>string</Mode>
         <Years>integer</Years>
      </DefaultRetention>
```

```
   </Rule>
</ObjectLockConfiguration>
```

Request Parameters

- `Bucket`
  The bucket for which you want to create or replace Object Lock configuration.
  Required: Yes
  Type: String

Request body

- `ObjectLockConfiguration`
  Root level tag for the `ObjectLockConfiguration` parameters.
  Required: Yes
- `ObjectLockEnabled`
  Indicates whether this bucket has an Object Lock configuration enabled. Enable `ObjectLockEnabled` when you apply `ObjectLockConfiguration` to a bucket.
  Valid Values: Enabled
  Required: No
  Type: String
- `Rule`
  Specifies the Object Lock rule for the specified objects. Enable the rule when you apply `ObjectLockConfiguration` to a bucket.
  The settings require both a mode and a period. The period can be either Days or Years. You cannot specify Days and Years at the same time.
  Required: No
  Type: `ObjectLockRule` data type

Response Syntax

```
HTTP/1.1 200
```

Possible Error Response

- `Success`
  HTTP status code 200.
- `AccessDenied`
  Access Denied.
  HTTP status code 403.
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.

- `InvalidBucketState`
  Object Lock configuration cannot be enabled on existing buckets.
  HTTP status code 409.
- `InvalidRequest`
  The error may occur for some reasons. For details, please refer the error messages.
  HTTP status code 400.

Get Object Lock Configuration (Flex WORM only)

Gets the Object Lock configuration for a bucket.

Request Syntax

```
GET /{bucket}?object-lock HTTP/1.1
Host: msdps3.server:8443
```

Request Parameters

- `Bucket`
  The bucket for which you want to retrieve Object Lock configuration.
  Required: Yes
  Type: String

Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ObjectLockConfiguration>
   <ObjectLockEnabled>string</ObjectLockEnabled>
   <Rule>
      <DefaultRetention>
         <Days>integer</Days>
         <Mode>string</Mode>
         <Years>integer</Years>
      </DefaultRetention>
   </Rule>
</ObjectLockConfiguration>
```

Response body

- `ObjectLockConfiguration`
  Root level tag for the ObjectLockConfiguration parameters.
  Required: Yes
- `ObjectLockEnabled`

Indicates whether this bucket has an Object Lock configuration enabled. Enable `ObjectLockEnabled` when you apply `ObjectLockConfiguration` to a bucket.

Valid Values: Enabled

Required: No

Type: String

- `Rule`

  Specifies the Object Lock rule for the specified objects. Enable the rule when you apply `ObjectLockConfiguration` to a bucket.

  The settings require both a mode and a period. The period can be either Days or Years. You cannot specify Days and Years at the same time.

  Required: No

  Type: `ObjectLockRule` data type

Possible Error Response

- `Success`

  HTTP status code 200.

- `AccessDenied`

  Access Denied.

  HTTP status code 403.

- `NoSuchBucket`

  The specified bucket does not exist.

  HTTP status code 404.

- `S3srvExtObjectLockConfigurationNotFound`

  Object Lock configuration does not exist for this bucket.

  HTTP status code 404.

- `InvalidRequest`

  The error may occur for some reasons. For details, please refer the error messages.

  HTTP status code 400.

S3 APIs on Objects

S3 APIs on objects perform the following main functions:

- Upload data (object) to the MSDP server.
- Download data from the MSDP server.
- Delete data from the MSDP server.
- List data in the MSDP server.

AbortMultipartUpload

Aborts a multipart upload. After a multipart upload is aborted, no additional parts can be uploaded using that upload ID. The storage that is consumed by any previously uploaded parts is freed.

Request Syntax

```
DELETE /bucket/Key+?uploadId=UploadId HTTP/1.1
Host: msdps3.server:8443
```

Request Parameters

- `Bucket`
  Name of the bucket.
  Required: Yes
  Type: String
- `Key`
  The name of the object for which multipart upload was initiated.
  Required: Yes
  Type: String
- `uploadId`
  Upload ID of multipart upload.
  Required: Yes
  Type: String

Response Syntax

```
HTTP/1.1 204
```

Possible Error Response

- `Success`
  HTTP status code 204.
- `AccessDenied`
  Request was rejected because user authentication failed.
  HTTP status code 403.
- `InvalidPrefix`
  Invalid Prefix.
  HTTP status code 400
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.

- `InternalError`
  Request failed because of an internal server error.
  HTTP status code 500.

CompleteMultipartUpload

Completes a multipart upload by assembling previously uploaded parts.

Request Syntax

```
POST /bucket/Key+?uploadId=UploadId HTTP/1.1
Host: msdps3.server:8443
<?xml version="1.0" encoding="UTF-8"?>
<CompleteMultipartUpload xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
<Part>
<ETag>string</ETag>
<PartNumber>integer</PartNumber>
</Part>
...
</CompleteMultipartUpload>
```

Request Parameters

- `Bucket`
  Name of the bucket.
  Required: Yes
  Type: String
- `Key`
  The name of the object.
  Required: Yes
  Type: String
- `uploadId`
  Upload ID of multipart upload.
  Required: Yes
  Type: String

Request body

- `CompleteMultipartUpload`
  Root level tag for the `CompleteMultipartUpload` parameters.
  Required: Yes
  - `Part`
    List of parts to create final object. It contains `ETag` and `PartNumber`.

- `ETag`
  `ETag` of the uploaded part.
- `PartNumber`
  `PartNumber` of the uploaded part.

Response Syntax

```
HTTP/1.1 200
x-amz-version-id: VersionId
<?xml version="1.0" encoding="UTF-8"?>
<CompleteMultipartUploadResult>
    <Bucket>string</Bucket>
    <Key>string</Key>
    <ETag>string</ETag>
</CompleteMultipartUploadResult>
```

Response Headers

- `x-amz-version-id`
  Version ID of the created object.

Response Body

- `CompleteMultipartUploadResult`
  Root level tag for the `CompleteMultipartUploadResult parameters`.
  Required: Yes
  - `Bucket`
    Name of the bucket.
    Required: Yes
    Type: String
  - `Key`
    The name of the object.
    Required: Yes
    Type: String
  - `ETag`
    SHA256 digest of the object.

Possible Error Response

- `Success`
  HTTP status code 200.
- `InvalidDigest`
  The Content-MD5 you specified did not match what was received.

HTTP status code 400.

- `InvalidArgument`
  Invalid Argument.
  Part number must be an integer between 1 and 10000, inclusive.
  HTTP status code 400.
- `InvalidPartOrder`
  The list of parts was not in ascending order.
  The parts list must be specified in order of the part number.
  HTTP status code 400.
  - `EntityTooLarge`
    Your proposed upload exceeds the maximum allowed object size.
    HTTP status code 400.
  - `ErrEntityTooSmall`
    Your proposed upload is smaller than the minimum allowed object size.
    HTTP status code 400.
  - `ErrNoSuchUpload`
    The specified multipart upload does not exist. The upload ID might not be valid, or the multipart upload might have been aborted or completed.
    HTTP status code 400.
  - `ErrInvalidPart`
    One or more of the specified parts could not be found. The part may not have been uploaded, or the specified entity tag may not match the part's entity tag.
    HTTP status code 400.
  - `MalformedPOSTRequest`
    The body of your POST request is not well-formed multipart/form-data.
    HTTP status code 400.
- `AccessDenied`
  Request was rejected because user authentication failed.
  HTTP status code 403.
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.
- `InternalError`
  Request failed because of an internal server error.
  HTTP status code 500.

CreateMultipartUpload

Initiates a multipart upload and returns an upload ID. This upload ID is used to associate all the parts in the specific multipart upload.

Request Syntax

```
POST /bucket/{Key+}?uploads HTTP/1.1
Host: msdps3.server:8443
```

Request Parameters

- `Bucket`
  Name of the bucket.
  Required: Yes
  Type: String
- `Key`
  The name of the object for which multipart upload was initiated.
  Required: Yes
  Type: String
- `x-amz-object-lock-mode` (Flex WORM only)
  Specifies the Object Lock mode that you want to apply to the uploaded object.
  Valid Values: GOVERNANCE, COMPLIANCE
- `x-amz-object-lock-retain-until-date` (Flex WORM only)
  Specifies the date and time when you want the Object Lock to expire.

  Note: If this option is not specified, the retention value will be calculated using the bucket default object lock configuration.

  ```
  object_lock_retain_until_date = current_system_timestamp +
  bucket_default_object_lock_retention
  ```

Response Syntax

```
HTTP/1.1 200
x-amz-version-id: VersionId
<?xml version="1.0" encoding="UTF-8"?>
<InitiateMultipartUploadResult>
    <Bucket>string</Bucket>
    <Key>string</Key>
    <UploadId>string</UploadId>
</InitiateMultipartUploadResult>
```

Response Body

- `InitiateMultipartUploadResult`
  Root level tag for the `InitiateMultipartUploadResult` parameters.
  Required: Yes
  - `Bucket`
    Name of the bucket.
  - `Key`
    The name of the object.
  - `UploadId`
    ID for the initiated multipart upload.

Possible Error Response

- `Success`
  HTTP status code 200.
- `InvalidArgument`
  Invalid Argument.
  HTTP status code 400.
- `AccessDenied`
  Request was rejected because user authentication failed.
  HTTP status code 403.
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.
- `InternalError`
  Request failed because of an internal server error.
  HTTP status code 500.
- `InvalidRequest`
  The error may occur for some reasons. For details, please refer the error messages.
  HTTP status code 400.

DeleteObject

Deletes the specified object in the bucket for a non-versioned bucket. If the versioning is enabled on the bucket and `VersionId` is passed, the specified version of the object is deleted. If the versioning is enabled on the bucket and `VersionId` is not passed, a `DeleteMarker` is created for the object.

Request Syntax

```
DELETE /bucket/Key+?versionId=VersionId HTTP/1.1
Host: msdps3.server:8443
```

Request Parameters

- `Bucket`
  Name of the bucket.
  Required: Yes
  Type: String
- `Key`
  The name of the object for which multipart upload was initiated.
  Required: Yes
  Type: String
- `versionId`
  The version ID of the Object.
  Type: String
- `x-amz-bypass-governance-retention` (Flex WORM only)
  Indicates whether S3 Object Lock should bypass Governance mode restrictions to process this operation. To use this header, you must have the `s3:BypassGovernanceRetention` permission.

Response Syntax

```
HTTP/1.1 204
x-amz-delete-marker: DeleteMarker
x-amz-version-id: VersionId
```

Response Headers

- `x-amz-delete-marker`
  Specifies if the deleted object is a delete marker or not.
- `x-amz-version-id`
  Specifies Version ID of the deleted object.

Possible Error Response

- `Success`
  HTTP status code 204.
- `InvalidArgument`
  Invalid Argument.
  HTTP status code 400.

- `AccessDenied`
  Request was rejected because user authentication failed.
  HTTP status code 403.
- `NoSuchKey`
  The specified key does not exist.
  HTTP status code 404.
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.
- `InternalError`
  Request failed because of an internal server error.
  HTTP status code 500.
- `InvalidRequest`
  Current object is protected by Object Lock and can not be overwritten.
  HTTP status code 400.

DeleteObjects

Deletes multiple objects from bucket by using a single request.

The Content-MD5 header is required for Multi-Object Delete request. S3 interface uses the header value to ensure that your request body has not been altered in transit.

Request Syntax

```
DELETE /bucket?delete HTTP/1.1
Host: msdps3.server:8443
<?xml version="1.0" encoding="UTF-8"?>
<Delete xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
    <Object>
        <Key>string</Key>
        <VersionId>string</VersionId>
    </Object>
    ...
    <Quiet>boolean</Quiet>
</Delete>
```

Request Parameters

- `Bucket`
  Name of the bucket.
  Required: Yes

Type: String

- `x-amz-bypass-governance-retention` (Flex WORM only)
  Indicates whether S3 Object Lock should bypass Governance mode restrictions to process this operation. To use this header, you must have the `s3:BypassGovernanceRetention` permission.

Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<DeleteResult>
    <Deleted>
        <DeleteMarker>boolean</DeleteMarker>
        <DeleteMarkerVersionId>string</DeleteMarkerVersionId>
        <Key>string</Key>
        <VersionId>string</VersionId>
    </Deleted>
    ...
    <Error>
        <Code>string</Code>
        <Key>string</Key>
        <Message>string</Message>
        <VersionId>string</VersionId>
    </Error>
    ...
</DeleteResult>
```

Response Body

- `DeleteResult`
  Root level tag for the `DeleteResult` parameters.
  Required: Yes
  - `Deleted`
    Information of the objects that are successfully deleted.
    - `DeleteMarker`
      Specifies if deleted object was a delete marker or not.
    - `DeleteMarkerVersionId`
      Specifies `versionId` of the deleted delete marker.
    - `Key`
      The name of the object
    - `VersionId`

`VersionId` of the deleted object.

- `Error`
  Information of the objects which failed to be deleted.
  - `Code`
    Error code of the error that occurred while deleting the object.
  - `Key`
    The name of the object
  - `Message`
    Error message
  - `VersionId`
    `VersionId` of the object or delete marker for which error occurred.

Possible Error Response

- `Success`
  HTTP status code 200.
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.
- `InternalError`
  Request failed because of an internal server error.
  HTTP status code 500.
- `InvalidRequest`
  Current object is protected by Object Lock and can not be overwritten.
  HTTP status code 400.

GetObject

Retrieves objects from an S3 bucket. For larger object download, use the range-based Get Object API.

Request Syntax

```
GET /bucket/Key+?partNumber=PartNumber&versionId=VersionId HTTP/1.1
Host: msdps3.server:8443
Range: Range
```

Request Parameters

- `Bucket`
  Name of the bucket.

Required: Yes
Type: String

- `Key`
  Name of the object.
  Required: Yes
  Type: String
- `partNumber`
  Number of the part of the object that is being read. This is a positive integer between 1 and 10,000.
  Type: Integer
- `versionId`
  Version Id of object.
  Type: String

Request Headers

- `Range`
  Returns the specified range bytes of object.
  Type: Integer

Response Syntax

```
HTTP/1.1 200
x-amz-delete-marker: DeleteMarker
accept-ranges: AcceptRanges
Last-Modified: LastModified
Content-Length: ContentLength
ETag: ETag
x-amz-version-id: VersionId
Content-Range: ContentRange
x-amz-storage-class: StorageClass
Body
```

Response Headers

- `x-amz-delete-marker`
  Specifies the object return is a delete marker or not. If the object is not a delete marker, this header does not get added in a response.
- `Last-Modified`
  The last modified time of the object.
- `Content-Length`
  Returned body size in bytes.

- `ETag`
  Specifies SHA256 of the returned object.
- `x-amz-version-id`
  Specifies the version ID of returned object.
- `Content-Range`
  The range of object that is returned in response.
- `x-amz-storage-class`
  Specifies the storage class of the returned object.
- `x-amz-object-lock-mode` (Flex WORM only)
  The Object Lock mode currently in place for this object.
  Valid Values: GOVERNANCE, COMPLIANCE
- `x-amz-object-lock-retain-until-date` (Flex WORM only)
  The date and time when this object's Object Lock expires.
- `x-amz-meta-msdps3-object-creator`
  The API used to upload the object. The value `PutGroupObject` means that "`PutObject` with the header `x-amz-meta-snowball-auto-extract`."
  Valid Values: `PutObject`, `PutGroupObject`, `UploadPart`

Possible Error Response

- `Success`
  HTTP status code 200.
- `InvalidArgument`
  Invalid Argument.
  Invalid version ID specified. HTTP status code 400.
- `EntityTooLarge`
  Your proposed upload exceeds the maximum allowed object size.
  HTTP status code 400.
- `AccessDenied`
  Request was rejected because user authentication failed.
  HTTP status code 403.
- `NoSuchKey`
  The specified key does not exist.
  HTTP status code 404.
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.

- `InternalError`

  Request failed because of an internal server error.
  HTTP status code 500.

HeadObject

Retrieves metadata from an object without returning the object itself. This operation is used when you are interested only in an object's metadata.

Request Syntax

```
HEAD /bucket/Key+?partNumber=PartNumber&versionId=VersionId HTTP/1.1
Host: msdps3.server:8443
```

Request Parameters

- `Bucket`

  Name of the bucket.
  Required: Yes
  Type: String

- `Key`

  Name of the object.
  Required: Yes
  Type: String

- `partNumber`

  Number of the part of the object that is being read. This is a positive integer between 1 and 10,000.
  Type: Integer

- `versionId`

  Version ID of object.
  Type: String

Response Syntax

```
HTTP/1.1 200
x-amz-delete-marker: DeleteMarker
accept-ranges: AcceptRanges
Last-Modified: LastModified
Content-Length: ContentLength
ETag: ETag
x-amz-version-id: VersionId
Content-Range: ContentRange
```

Response Headers

- `x-amz-delete-marker`
  Specifies the object return is a delete marker or not. If the object is not a delete marker, this header does not get added in a response.
- `Last-Modified`
  The last modified time of the object.
- `Content-Length`
  Returned body size in bytes.
- `ETag`
  Specifies SHA256 of returned object.
- `x-amz-version-id`
  Specifies the version ID of returned object.
- `Content-Range`
  The range of object that is returned in response.
- `x-amz-object-lock-mode` (Flex WORM only)
  The Object Lock mode currently in place for this object.
  Valid Values: GOVERNANCE, COMPLIANCE
- `x-amz-object-lock-retain-until-date` (Flex WORM only)
  The date and time when this object's Object Lock expires.
- `x-amz-meta-msdps3-object-creator`
  The API used to upload the object. The value `PutGroupObject` means that "`PutObject` with the header `x-amz-meta-snowball-auto-extract`."
  Valid Values: `PutObject`, `PutGroupObject`, `UploadPart`

Possible Error Response

- `Success`
  HTTP status code 200.
- `InvalidArgument`
  Invalid Argument. Invalid version ID specified.
  HTTP status code 400.
- `AccessDenied`
  Request was rejected because user authentication failed.
  HTTP status code 403.
- `NoSuchKey`
  The specified key does not exist.
  HTTP status code 404.
- `NoSuchBucket`

The specified bucket does not exist.
HTTP status code 404.

- `InternalError`
  Request failed because of an internal server error.
  HTTP status code 500.

PutObject

Adds an object to a bucket. If the bucket is enabled for versioning, Put Object API returns the `VersionId` of the object.

Request Syntax

```
PUT /bucket/Key HTTP/1.1
Host: msdps3.server:8443
Content-Length: ContentLength
Content-MD5: ContentMD5
Body
```

Request Parameters

- `Bucket`
  Name of the bucket.
  Required: Yes
  Type: String
- `Key`
  Name of the object.
  Required: Yes
  Type: String
- `x-amz-object-lock-mode` (Flex WORM only)
  The Object Lock mode that you want to apply to this object.
  Valid Values: GOVERNANCE, COMPLIANCE
- `x-amz-object-lock-retain-until-date` (Flex WORM only)
  The date and time when you want this object's Object Lock to expire. Must be formatted as a timestamp parameter.

Response Syntax

```
HTTP/1.1 200
ETag: ETag
x-amz-version-id: VersionId
```

Response Headers

- `x-amz-version-id`
  The version ID of the object PUT in the bucket.

Possible Error Response

- `Success`
  HTTP status code 200.
- `EntityTooLarge`
  The object size exceeded maximum allowed size.
  HTTP status code 400.
- `AccessDenied`
  Request was rejected because user authentication failed.
  HTTP status code 403.
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.
- `InternalError`
  Request failed because of an internal server error.
  HTTP status code 500.
- `InvalidRequest`
  The error may occur for some reasons. For details, please refer the error messages.
  HTTP status code 400.

Copy Object

Creates a copy of an object in the storage server. You must have read access to the source object and write access to the destination bucket. If bucket is versioning enabled, then Copy Object API returns the `VersionId` of the object. When copying an object, both the metadata and the ACLs are not preserved.

Request Syntax

```
PUT /bucket/Key HTTP/1.1
Host: msdps3.server:8443
x-amz-copy-source: CopySource
x-amz-object-lock-mode: ObjectLockMode
x-amz-object-lock-retain-until-date: ObjectLockRetainUntilDate
```

Request Parameters

- `Bucket`
  The name of the destination bucket.

Required: Yes
Type: String

- `Key`
  The key of destination object.
  Required: Yes
  Type: String

- `x-amz-copy-source`
  Specifies the source object for the copy operation.
  The value format: Specify the name of the source bucket and the key of the source object, separated by a slash (/).
  For example, to copy the object `msdps3/copyright.txt` from the bucket **srcbk**, use `srcbk/msdps3/copyright.txt`. The value must be URL-encoded.
  To copy a specific version of an object, append `?versionId=<version-id>` to the value. For example,
  `srcbk/msdps3/copyright.txt?versionId=AAAA1234567890`
  If you don't specify a version ID, the latest version of the source object is copied.
  Pattern: `\/.+\/.+`
  Required: Yes

- `x-amz-object-lock-mode` (Flex WORM only)
  The Object Lock mode that you want to apply to this copied object.
  Valid Values: GOVERNANCE, COMPLIANCE

- `x-amz-object-lock-retain-until-date (Flex WORM only)`
  The date and time when you want this copied object's Object Lock to expire. It must be formatted as a timestamp parameter.

Response Syntax

```
HTTP/1.1 200
x-amz-copy-source-version-id: CopySourceVersionId
x-amz-version-id: VersionId
<?xml version="1.0" encoding="UTF-8"?>
<CopyObjectResult>
   <ETag>string</ETag>
   <LastModified>timestamp</LastModified>
</CopyObjectResult>
```

Response Headers

- `x-amz-copy-source-version-id`
  Version of the copied object in the source bucket.

- `x-amz-version-id`

Version ID of the newly created copy.

- `ETag`
  Returns the ETag of the new object.
  Type: String
- `LastModified`
  Creation date of the object.
  Type: Timestamp

Possible Error Response

- `Success`
  HTTP status code 200.
- `EntityTooLarge`
  The object size exceeded maximum allowed size.
  HTTP status code 400.
- `AccessDenied`
  Request was rejected because user authentication failed.
  HTTP status code 403.
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.
- `InternalError`
  Request failed because of an internal server error.
  HTTP status code 500.
- `InvalidRequest`
  The error may occur for some reasons. For details, please refer the error messages.
  HTTP status code 400.

UploadPart

Uploads a part in a multipart upload.

Request Syntax

```
PUT /bucket/Key+?partNumber=PartNumber&uploadId=UploadId HTTP/1.1
Host: msdps3.server:8443
```

Request Parameters

- `Bucket`
  Name of the bucket.

Required: Yes
Type: String

- `Key`
  Name of the object.
  Required: Yes
  Type: String
- `partNumber`
  Number of the part that is being uploaded.
  Required: Yes
  Type: String
- `uploadId`
  Upload ID of multipart upload.
  Required: Yes
  Type: String
- `Content-MD5`
  The base64-encoded 128-bit MD5 digest of the part data. This parameter is required if object lock parameters are specified.

Response Syntax

```
HTTP/1.1 200
```

Possible Error Response

- `Success`
  HTTP status code 200.
- `InvalidArgument`
  HTTP status code 400.
- `AccessDenied`
  Request was rejected because user authentication failed.
  HTTP status code 403.
- `NoSuchUpload`
  The upload ID or Key might be invalid.
  HTTP status code 404.
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.
- `InternalError`
  Request failed because of an internal server error.

HTTP status code 500.

PutObject (snowball-auto-extract for small files)

Each copy operation has some overhead; therefore, performing many transfers on individual small files has slower overall performance than transferring the same data in larger files. To significantly improve your transfer speed for small files (files less than 1 MB), batch the small files together. Batching files is a manual process. If the batched files are put to S3 server with the `x-amz-meta-snowball-auto-extract` header, the batches are automatically extracted when data is imported MSDP S3 Server.

Note: The `x-amz-meta-snowball-auto-extract` header is not accepted for un-versioned bucket, and all the batched small files share the same version in S3 server.

Run the tar or gzip command to manually batch small files, and then transfer them to S3 interface for MSDP.

For example: `tar -czf <archive-file> <small files or directory of small files>`

```
aws --endpoint https://<hostname>:8443 --profile <profile name> s3api
[--ca-bundle <CA_BUNDLE_FILE>] put-object --bucket <bucket name>
--key <key path> --body <xxx.tgz> --metadata
snowball-auto-extract=true
```

Keep in mind the following when batching small files:

- Maximum batch size of 5 GB.
- Recommended maximum of 10,000 files per batch.
- Supported archive formats are TGZ.

Request Syntax

```
PUT /bucket/Key HTTP/1.1
Host: msdps3.server:8443
Content-Length: ContentLength
Content-MD5: ContentMD5
x-amz-meta-snowball-auto-extract:true
Body
```

Request Parameters

- `Bucket`

Name of the bucket.
Required: Yes
Type: String

- `Key`
  Name of the object.
  Required: Yes
  Type: String
- `x-amz-object-lock-mode` (Flex WORM only)
  The Object Lock mode that you want to apply to this object.
  Valid Values: GOVERNANCE, COMPLIANCE
- `x-amz-object-lock-retain-until-date` (Flex WORM only)
  The date and time when you want this object's Object Lock to expire. Must be formatted as a timestamp parameter.

Request Headers

- `Enable snowball-auto-extract`
  Required: Yes
  Value: true

Response Syntax

```
HTTP/1.1 200
ETag: ETag
x-amz-version-id: VersionId
```

Request Headers

- `x-amz-version-id`
  The version-id of the object PUT in the bucket.

Possible Error Response

- `Success`
  HTTP status code 200.
- `EntityTooLarge`
  The object size exceeded maximum allowed size.
  HTTP status code 400.
- `AccessDenied`
  Request was rejected because user authentication failed.
  HTTP status code 403.
- `NoSuchBucket`
  The specified bucket does not exist.

HTTP status code 404.

- `InternalError`
  Request failed because of an internal server error.
  HTTP status code 500.
- `InvalidRequest`
  The error may occur for some reasons. For details, please refer the error messages.
  HTTP status code 400.

Put Object Retention (Flex WORM only)

Places an Object Retention configuration on an object.

Request Syntax

```
PUT /{bucket}/{Key+}?retention&versionId=VersionId HTTP/1.1
Host: msdps3.server:8443
x-amz-bypass-governance-retention: BypassGovernanceRetention
Content-MD5: ContentMD5
<?xml version="1.0" encoding="UTF-8"?>
<Retention xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
   <Mode>string</Mode>
   <RetainUntilDate>timestamp</RetainUntilDate>
</Retention>
```

Request Parameters

- `Bucket`
  The bucket name that contains the object you want to apply this Object Retention configuration to.
  Required: Yes
  Type: String
- `Key`
  The key name for the object that you want to apply this Object Retention configuration to.
  Required: Yes
  Type: String
- `versionId`
  The version ID for the object that you want to apply this Object Retention configuration to.
- `x-amz-bypass-governance-retention`
  Indicates whether this action should bypass Governance mode restrictions.

Request Body

- `Retention`
  Root level tag for the Retention parameters.
  Required: Yes
- `Mode`
  Indicates the Retention mode for the specified object.
  Valid Values: GOVERNANCE, COMPLIANCE
  Required: No
  Type: String
- `RetainUntilDate`
  The date on which this Object Lock Retention will expire.
  Required: No
  Type: Timestamp

Response Syntax

```
HTTP/1.1 200
```

Possible Error Response

- `Success`
  HTTP status code 200.
- `AccessDenied`
  Request was rejected because user authentication failed.
  HTTP status code 403.
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.
- `InternalError`
  Request failed because of an internal server error.
  HTTP status code 500.
- `InvalidRequest`
  The error may occur for some reasons. For details, please refer the error messages.
  HTTP status code 400.

Get Object Retention (Flex WORM only)

Retrieves an object's retention settings.

Request Syntax

```
GET /{bucket}/Key+?retention&versionId=VersionId HTTP/1.1 Host: msdps3.server
```

Request Parameters

- `Bucket`
  The bucket name that contains the object for which you want to retrieve the retention settings.
  Required: Yes
  Type: String
- `Key`
  The key name for the object for which you want to retrieve the retention settings.
  Required: Yes
  Type: String
- `versionId`
  The version ID for the object for which you want to retrieve the retention settings.
  Type: String

Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<Retention>
   <Mode>string</Mode>
   <RetainUntilDate>timestamp</RetainUntilDate>
</Retention>
```

Response Body

- `Retention`
  Root level tag for the Retention parameters.
  Required: Yes
- `Mode`
  Indicates the Retention mode for the specified object.
  Valid Values: GOVERNANCE, COMPLIANCE
  Type: String
- `RetainUntilDate`
  The date on which this Object Lock Retention expires.
  Type: Timestamp

Possible Error Response

- `Success`
  HTTP status code 200.
- `InvalidArgument`
  Invalid Argument. Invalid version id specified.
  HTTP status code 400.
- `AccessDenied`
  Request was rejected because user authentication failed.
  HTTP status code 403.
- `NoSuchKey`
  The specified key does not exist.
  HTTP status code 404.
- `NoSuchBucket`
  The specified bucket does not exist.
  HTTP status code 404.

The naming rules for buckets and objects

The naming rules for buckets:

- Bucket name must be between 3 and 63 characters long.
- Bucket name can consist only lowercase letters, numbers, dots (.) and hyphens (-).
- Bucket name must begin and end with a letter or number.
- Bucket name must not be an IP address. For example, 192.168.5.4.
- Bucket name must not start with the prefix **xn--**.
- Bucket name must not end with the suffix **-s3alias**.
- Avoid using dots (.) in bucket names.

The naming rules for objects:

- The object name rules are the same as the file naming in UNIX file system.
- Only URL encoding type is supported. The name should be encoded by URL Escape if escape is needed.
- The object name must not start with or end with the slash.
- The object name is processed by the following rules:
  - Replace multiple slashes with a single slash.
  - Eliminate each . path name element (the current directory).

- Eliminate each inner .. path name element (the parent directory) along with the non-.. element that precedes it.
- Eliminate .. elements that begin a rooted path: that is, replace "/.." by "/" at the beginning of a path.
- The returned path ends in a slash only if it is the root "/".
- If the result of this process is an empty string, it returns the string ".".
- If the object name includes "%", it is treated as encoded name.

Creating a protection policy for the MSDP object store

The MSDP-Object-Store policy type protects the data that is put in MSDP S3 storage.

Following are the limitations about MSDP object store backups:

- The media server must be a Red Hat operating system.
- If bucket versioning is enabled, only the latest object versions can be backed up.
- Only supports the MSDP S3 configured using NBCA and ECA.
- You can provide multiple backup selections in one backup policy. However, these backup selections must belong to a same bucket.
- If the MSDP S3 path you provide matches both an object and a prefix, the backup job fails.
  For example, if the path is provided as `/bucket1/obj1` and the following object structure exists, the backup job fails: /bucket1/obj1, /bucket1/obj1/obj2.
- When MSDP object store backup job is running, the objects that are being backed up cannot be deleted or overwrote (non-versioning bucket) until the backup jobs finishes.

To create a protection policy for the MSDP object store

1. Create a policy using the NetBackup web UI.
2. On the **Attributes** tab, select **MSDP-Object-Store** from the policy type list.
3. Under **Destination**, select storage unit from the policy storage list.

   The storage unit for MSDP-Object-Store policy must be in the same disk pool volume where the MSDP S3 bucket is created.

   See Policy storage (policy attribute) in the *NetBackup Administrator's Guide Volume I* for more information about policy storage setting.

4 On the **Schedules** tab, select Full backup, Differential incremental backup, or Cumulative incremental backup.

5 On the **Backup Selections** tab, click **Add**. Enter the path of the MSDP S3 bucket prefix in Pathname. Format: `/<BucketName>/<Prefix>`. Click **Add to list**.

The client name of the MSDP-Object-Store policy is generated automatically using the format `MSDP_<StorageServerName>_<BucketName>`.

6 Run the MSDP-Object-Store policy.

After the backups are created, you can manage the backups with NetBackup features, such as restore, duplication, Auto Image Replication, and so on.

About MSDP S3 object import

In earlier NetBackup releases, a replication storage lifecycle policy (SLP) for MSDP-Object-Store backups only replicates the NetBackup backup image and imports it into the replica domain's NetBackup catalog. The underlying S3 objects themselves were not synchronized between MSDP S3 buckets on the two domains.

MSDP S3 object import lets you keep a production MSDP S3 bucket and a disaster recovery replica bucket in sync. Data that is ingested into an MSDP S3 bucket on one MSDP instance (a Flex appliance or build-your-own MSDP server) is synchronized to a bucket of the same name on another MSDP instance in a replica domain, using your existing AIR/SLP replication. As a result, failover clients see the same objects in the replica bucket without needing to run a restore.

MSDP S3 object import reuses the existing Auto Image Replication (AIR) and storage lifecycle policy flow; it does not require a separate replication mechanism. When an MSDP-Object-Store backup image is replicated to the replica domain, NetBackup also triggers an import operation that reconstructs the bucket's objects on the replica MSDP S3 server from the replicated image's object catalog. The same flow works in the reverse direction, so either domain can act as source or target.

Note: Synchronization between the production and replica MSDP S3 buckets is not real-time. Objects appear in the replica bucket only after the backup and replication jobs for that image have completed. Avoid writing new objects into a bucket while a backup, replication, or import operation is in progress on that bucket.

Supported platforms: MSDP build-your-own (BYO) server, NetBackup Flex Appliance.

See “About S3 interface for MSDP ” on page 415.

See “Creating a protection policy for the MSDP object store” on page 503.

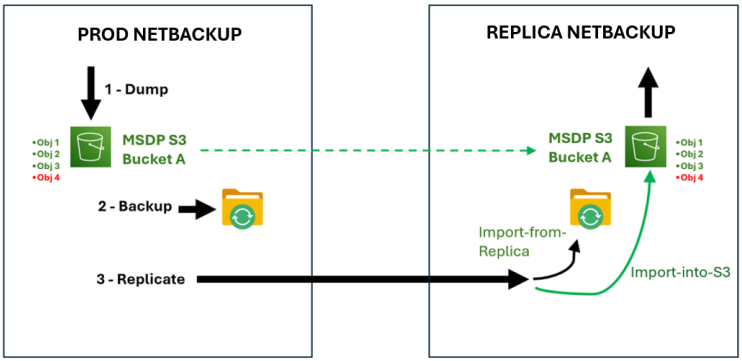
See “About MSDP replication to a different domain” on page 136.

How MSDP S3 object import works

The following steps describe how MSDP S3 object import synchronizes objects between the production and replica domains.

1. In the production domain, an MSDP-Object-Store backup policy runs and NetBackup creates a backup image of the S3 bucket's objects.
2. NetBackup starts a replication job, through the configured AIR SLP, to replicate the image to the replica domain.
3. When the replication job finishes, the production MSDP storage server generates an additional import event in the replica domain, alongside the standard AIR image import event. This event carries the information NetBackup needs to import the replicated image into the replica MSDP S3 catalog.
4. In the replica domain, NetBackup polls for and picks up the import S3 event, and creates an import into S3 bucket job. This job calls the MSDP S3 server to start importing the object metadata.
5. The MSDP S3 server reads the object listing captured in the replicated backup image and loads that metadata into the S3 service, making the objects available in the replica bucket.
6. NetBackup periodically polls the MSDP S3 server until the import completes, and updates the job status in the Activity Monitor.
7. This same sequence can run in the opposite direction, importing objects from the replica domain back into the production domain's bucket, provided that S3 import is also enabled for the corresponding policy on the replica MSDP storage server.

Figure 8-1 How S3 object import works



Prerequisites for MSDP S3 object import

Ensure that the prerequisites for S3 interface for MSDP are met, along with the following additional prerequisites, to use MSDP-S3 object import:

See "Prerequisites for MSDP build-your-own (BYO) server" on page 416.

See "Instant access for MSDP object store" on page 510.

- The MSDP S3 bucket name and the MSDP-Object-Store backup policy name must be identical on the production and replica domains.
- A storage lifecycle policy that performs Backup, Replication, and Import (in that order) must be configured.
- MSDP S3 import must be explicitly enabled for the backup policy on the source MSDP storage server, using the `cacontrol --s3import` command.

Note: These prerequisites are common for Flex and BYO.

Configuring MSDP S3 object import

This section describes the steps to configure MSDP S3 object import between a production and a replica domain.

To configure MSDP S3 object import

1 In the production domain:

- Configure an AIR storage lifecycle policy that replicates to the replica domain.
  See “Creating a storage lifecycle policy” on page 165.
- Configure an MSDP-Object-Store backup policy that uses this AIR SLP.
  See “Creating a protection policy for the MSDP object store” on page 503.

2 In the replica domain:

- Configure an AIR storage lifecycle policy that replicates back to the production domain.
- Configure an MSDP-Object-Store backup policy with the same policy name as the one created in the production domain, using the same S3 bucket name as the production domain bucket.

Note: This step is optional. Configure it when the replica domain is expected to take over as the production domain after a disaster recovery event, so that objects created there can be synchronized back once the original production domain is recovered.

---

3 On the source MSDP storage server (for production-to-replica S3 object syncs, enable the import event for the policy:

```
cacontrol --s3import updatebypolicy <PolicyName>
IMPORT_V1
```

4 Repeat step 3 on the replica domain's storage server if you also want objects to sync in the reverse direction (replica to production).

5 Write objects into the S3 bucket in the production domain.

6 Run the MSDP-Object-Store backup policy. NetBackup creates the backup image, replicates it, and triggers the import automatically on the replica domain. No additional manual step is required after configuration.

Note: Configure `cacontrol --s3import` only on the source MSDP storage server for a given replication direction.

---

cacontrol --s3import command options

Use `cacontrol --s3import` to control which MSDP-Object-Store backup policies generate an S3 import event during AIR replication. Run this command on the source MSDP storage server only.

Table 8-6

Action	Description	Command
Enable	Enables the import event for the policy.	`cacontrol --s3import updatebypolicy <PolicyName> IMPORT_V1`
Check status	Reports whether import is enabled for the policy.	`cacontrol --s3import getbypolicy <PolicyName>`
Disable	Disables the import event for the policy.	`cacontrol --s3import deletebypolicy <PolicyName>`

Limitations of the cacontrol --s3import command

The `cacontrol --s3import` command has the following limitations.

- The command does not validate the policy name. It does not verify that the policy exists or that it is an MSDP-Object-Store policy. A marker can be set against any policy name.
- An import event is generated only for MSDP-Object-Store policies at replication time. If a marker is set on a non-MSDP-S3 policy, no import event is created.
- Only IMPORT_V1 is currently supported as the marker value.
- Do not edit `marker.cfg` manually. Always use `cacontrol --s3import` command.

Viewing MSDP S3 import jobs

After a replicated MSDP-Object-Store backup image finishes replicating, an **Import into S3 bucket** job starts on the replica site, in addition to the existing Import from replica job. Both are visible in the Activity Monitor.

The **Import into S3 bucket** job:

- Requests the MSDP S3 server to start the import task.
- Reads the object metadata from the replicated MSDP-Object-Store backup image.
- Writes that object metadata into the MSDP S3 bucket on the replica.
- Is tracked and updated in the Activity Monitor until the import completes.

You can also trigger and check catalog import through the NetBackup REST API.

Limitations of MSDP S3 object import

MSDP S3 object import has the following limitations.

- You cannot back up or dump new objects into a bucket while an import operation is in progress for that bucket.
- Only one import operation at a time is allowed per bucket. Concurrent imports for the same bucket are serialized.
- If an import fails due to a transient error, it is retried automatically. A retry starts the import over from the beginning rather than resuming.
- Only replicated MSDP-Object-Store backups can be imported on the remote site. You cannot import a local backup on the same site where it was created.
- You cannot retrieve objects from a bucket while an Import into S3 bucket job is in progress for that bucket. This is expected behavior as object retrieval is not allowed on a bucket during import to avoid returning inconsistent data.
- If an import fails more than the maximum configured retry count (`CATALOG_IMPORT_FAILED_MAX_RETRIES`), NetBackup stops trying to import that specific backup image - neither the automatic replication workflow nor a manual import through the API can import it after that. Import resumes normally with the next replicated backup for the same bucket and policy.

Troubleshooting MSDP S3 object import

If an import job is not triggered on the replica site after replication completes:

1. Confirm the import marker is enabled for the policy on the source MSDP storage server:

   ```
   cacontrol --s3import getbypolicy <PolicyName>
   ```

2. Check the following logs on the MSDP storage server:
   - cata2s3.log
   - s3srv.log

A successful import logs entries similar to the following in s3srv.log:

```
info s3server/gateway.(*LsuInfoSys).startIAShareMonitor:
import job detected for bucket
'<bucket-name>', starting share processing under exclusive
import lock
...
info s3server/gateway.(*instantAccessManager).startTimer:
```

```
all the files in this share
have been processed. succeeded(1), failed(0), total(1)
```

Recovering the MSDP object store data from the backup images

You can recover the backup data from MSDP-Object-Store backup images and write to a NetBackup client's file system.

Limitations about the MSDP object store recovery:

- Supports recovering the MSDP object store backup data to a NetBackup client's file system only, and the NetBackup client's operating system must be Linux.
- To be more secure, the recovered objects' permission is 600 in the file system, which needs administrator privilege to access them.

To recover the MSDP-Object-Store backup data

1. On the left, click **Recovery** in the NetBackup web UI.
2. Under **Regular recovery**, click **Start recovery**.
3. On the **Basic properties** page, select **MSDP-Object-Store** from the policy type list.
4. Click **Select client** to select the source client.
5. Type the destination client host name. The destination client must be a valid NetBackup client. The automatically generated client name for MSDP-Object-Store policy cannot be used as the destination client.

   Click **Next**.
6. On the **Recovery details** page, select backup images and objects to be recovered. Click **Next**.
7. On the **Recovery options** page, select the Destination of the recovery. This is a file system path of the destination client. Click **Next**.
8. On the **Review** page, review the details and click **Start recovery**.

   After the restore job is finished, the recovered objects are in the destination file system path of the destination NetBackup client.

Instant access for MSDP object store

You can create instant access buckets for MSDP object store using NetBackup images.

Prerequisites of instant access

- The MSDP object store instant access is only applicable for Red Hat MSDP storage servers with Build-Your-Own (BYO) and NetBackup Flex Appliance environment.
- You must generate at least one backup image for MSDP object store on the MSDP S3 bucket.
- MSDP S3 interface must be configured on MSDP storage where the NetBackup images are located.

Things to consider before you use the instant access feature

Note the following about the instant access MSDP object store features:

- The instant access target storage server must have the MSDP S3 interface configured.
  See "About S3 interface for MSDP " on page 415.
- Versioning must be disabled on the instant access target bucket. If you specify a new bucket, by default versioning is disabled. If you specify an existing bucket where versioning is enabled, instant access fails.
- The instant access target bucket must be located on the storage where the backup image is stored. If you specify a new bucket, by default it is created on the same storage where the backup image is stored. If you specify an existing bucket that is on a storage server that is different from where the backup image is stored, instant access fails.
- For the source MSDP S3 buckets, if the backup images are generated in the earlier NetBackup releases, they may not be identified for instant access. You must generate backup images from the latest release to use MSDP object store instant access.
- It is not recommended to use MSDP object store backup and instant access at the same time on the same MSDP storage server.
- You must use the MSDP-Object-Store policy to create a NetBackup backup image.

Create an instant access MSDP object store

You can create an instant access MSDP object store from a NetBackup image. The **Buckets** tab displays the source buckets and the **Instant access MSDP object store** tab displays all the target buckets that you create.

Note: The instant access for MSDP object store feature only supports BYO and NetBackup Flex media platform.

Things to consider before you use the instant access feature

To create an instant access MSDP object store

1. On the left, click **Workloads > MSDP object store**.
2. Locate the bucket on the **Buckets** tab and click on it.
3. Click the **Recovery points** tab, then click the date on which the backup occurred.

   The available images appear in rows with the backup timestamp for each image.
4. On the image or the copy of the image that has the option to recover using instant access, click **Recover**.

   On the **Create an instant access MSDP object store** page, the MSDP object storage server name is displayed.
5. In the **Target bucket name** field enter the name of the bucket where you want to create the instant access MSDP object store, and then click **Save**.

   The target bucket name must be between 3 and 63 characters, must begin and end with a letter or number, and can contain only lowercase letters, numbers dots, and hyphens.
6. After the instant access job starts, you can click the **Restore activity** tab to view the progress.
7. For details about the target bucket, click on the bucket display name under the **Instant access MSDP object store** tab.

Delete an instant access MSDP object store

You must run the following command line options on the MSDP storage server where the instant access target bucket is located. This operation is not supported for the NetBackup web UI.

- `aws --cli-read-timeout 0 --endpoint <endpoint url> s3 rm s3://<bucket name> --recursive` (empty the bucket)
- `aws --endpoint <endpoint url> s3 rb s3://<bucket name>` (remove the bucket)
- `rm -f <storage dir>/s3_ia_instances/<bucketname>.json`

Disaster recovery in S3 interface for MSDP

After you recover the local LSU, run `s3srv_config.sh --recover` command to recover S3 Interface for MSDP.

See "Recovering from an MSDP storage server disk failure" on page 583.

See "Recovering from an MSDP storage server failure" on page 584.

You can recover cloud LSU in MSDP cloud. See "About the disaster recovery for cloud LSU" on page 322.

As for Scenario 1, Scenario 2, Scenario 5, and Scenario 6(local storage is lost), run S3 configuration command to configure S3.

As for Scenario 3 and Scenario 4 (local storage is not lost), run `s3srv_config.sh --recover` command to configure S3.

Recovering the MSDP S3 IAM configurations from cloud LSU

If MSDP S3 is enabled, you can recover the MSDP S3 IAM configurations from the cloud LSU that is recovered from the disaster.

To recover the MSDP S3 IAM configurations from cloud LSU

- Run the following command on the MSDP server.

  ```
  /usr/openv/pdde/vxs3/cfg/script/s3srv_config.sh
  --recover-iam-config <LSU name>
  ```

  The command displays the IAM configurations in the cloud LSU and current IAM configurations.

  The following warning appears:

  ```
  WARNING: This operation overwrites current IAM configurations
  with the IAM configurations in cloud LSU.
  ```

  To overwrite the current IAM configurations, type the following and press **Enter**.

  ```
  overwrite-with-<cloud_LSU_name>
  ```

Note: Do not run `/usr/openv/pdde/vxs3/cfg/script/s3srv_config.sh --reset-iam-root` command before this command. It overwrites the IAM configurations in the cloud LSU.

Limitations in S3 interface for MSDP

S3 interface for MSDP has the following limitations:

- 1000 buckets in one S3 server.
- Recommend 500~800 concurrent requests based on your MSDP server performance.
- Small file must be less than 1 MB. A tar compress file of small files must be less than 5 GB. The number of small files in small files tar compress file should be less than 10000.
- One object size must be less than 5 TB.
- Multipart upload limitation. See Amazon S3 multipart upload limits
- The S3 interface for MSDP does not support Amazon S3 Glacier, Deep Archive, and Microsoft Azure Archive.
- When you configure S3 server, the client's name in the OST plug-in should avoid #s3storage.
- Before you change the back-end WORM mode, ensure that all the multipart uploads are completed.
- Objects cannot be copied across different storage servers.
- Objects greater than 5 GB cannot be copied.
- `UploadPartCopy` API is not supported. If you copy the objects that are uploaded by `UploadPart`, the created objects will be single objects without any parts.
- The S3 interface for MSDP does not support AWS Command Line Interface version 2.23.0 or later as these versions use chunked uploads with trailers for PUT actions such as `PutObject` and `UploadPart`.

Logging and troubleshooting

S3 interface for MSDP log is saved at `<storage>/log/vxs3`. You can also find the errors that are related to S3 API at this location. Some errors can be found under `spad/spoold`.

Configuring the logging

1. Configure the log level manually.

   Edit S3 server configuration file `<storage>/etc/puredisk/s3srv.cfg`

   ```
   ; None: 0; Error: 1; Warning: 2; Info: 3; Debug: 4

    ; @restart

    LogLevel=<log level>
   ```

2. Restart the S3 server.

   ```
   systemctl restart pdde-s3srv
   ```

NGINX forwards request to the S3 server. NGINX logging is disabled by default. If you want to use it, you must enable NGINX log in S3 NGINX configuration.

Enabling NGINX logging

1. Edit `/etc/<nginx server name>/conf.d/s3srvbyo.conf`.
2. Change access and error part as follows:

   ```
    access_log <access log path> main;

           error_log <error log path> debug;
   ```

3. Reload the NGINX configuration.

   ```
   systemctl reload <nginx server name>
   ```

Best practices

Following are the best practices for using S3 interface for MSDP:

- Use prefix in object keys.
- Avoid using dots (.) in bucket names.
- Recommend 50~100 concurrent requests based on BYO computer performance.
- Recommend less than 1000 `CommonPrefixes` elements under a specified prefix that are separated by the slash (/) delimiter character.
- When you use AWS CLI, add `--cli-read-timeout 0` and `--cli-connect-timeout 0` and add `payload_signing_enabled = true` in `.aws/config` file.

- The time-out settings in S3 client can affect the request procedure. When the server does not respond before the time-out, the client cancels the request automatically.

Chapter 9

Monitoring deduplication activity

This chapter includes the following topics:

- Monitoring the MSDP deduplication and compression rates
- Viewing MSDP job details
- About MSDP storage capacity and usage reporting
- About MSDP container files
- Viewing storage usage within MSDP container files
- About monitoring MSDP processes
- Reporting on Auto Image Replication jobs
- Checking the image encryption status
- MSDP events in NetBackup web UI

Monitoring the MSDP deduplication and compression rates

The deduplication rate is the percentage of data that was stored by the deduplication engine. That data is not stored again. The compression rate is the percentage of space that is saved by compressing the backup data before it is stored.

The following methods show the MSDP deduplication rate:

- To view the global MSDP deduplication ratio
- To view the MSDP deduplication ratio for a backup job in the Activity monitor

For the method to show the MSDP compression rate, See "Viewing MSDP job details" on page 518.

On UNIX and Linux, you can use the NetBackup `bpdbjobs` command to display the deduplication rate. However, you must configure it to do so.

To view the global MSDP deduplication ratio

1. Open the web UI.
2. On the left, click **Storage > Disk storage**.
3. Click the **Storage servers** tab.
4. Click the storage server name to view the global MSDP deduplication ratio.

To view the MSDP deduplication ratio for a backup job in the Activity monitor

1. In the **NetBackup web UI**, click **Activity monitor**.
2. Click the **Jobs** tab.

   The **Deduplication ratio** column shows the ratio for each job.

Disable the display of separate deduplication and compression rates

To disable the display of the compression rate separately:

- Open the `pd.conf` file that is available at the following locations:

  Windows

  `<install_location>\lib\ost-plugins\pd.conf`

  UNIX

  `/usr/openv/lib/ost-plugins/pd.conf`

- Add the following parameter in the file:

  `DISPLAY_COMPRESSION_SPACE_SAVING = 0`

  You can remove this parameter or change the value to `1` to turn on the display of compression rate as a separate value.

See "About MSDP server requirements" on page 37.

See "About MSDP client deduplication requirements and limitations" on page 40.

Viewing MSDP job details

Use the NetBackup Activity Monitor to view deduplication job details.

To view MSDP job details

1. In the **NetBackup web UI**, click **Activity Monitor**.
2. Click the **Jobs** tab.
3. To view the details for a specific job, double-click on the job that is displayed in the **Jobs** tab pane.
4. In the **Job Details** dialog box, click the **Detailed Status** tab.

   The deduplication job details are described in a different topic.

   See "MSDP job details" on page 519.

MSDP job details

In the Activity monitor, the job details show the details of a deduplication job. The details depend on whether the job is media server deduplication or client-side deduplication job.

Media server deduplication job details

For media server deduplication, the **Details** tab shows the deduplication rate on the server that performed the deduplication. The following job details excerpt shows details for a client for which MSDP_Server.example.com deduplicated the data (the **dedup** field shows the deduplication rate and the **compression** field shows the storage space that is saved by compression):

```
LOG 1551428319 4 Info MSDP_Server.example.com 27726
StorageServer=PureDisk:MSDP_Server.example.com; Report=PDDO Stats
(multi-threaded stream used) for (MSDP_Server.example.com):
scanned: 105098346 KB, CR sent: 2095410 KB, CR sent over FC: 0 KB,
dedup: 98.0%, cache hits: 337282 (41.0%), where dedup space saving:89.7%,
compression space saving:8.3%
```

Client-side deduplication job details

For client-side deduplication jobs, the **Details** tab shows two deduplication rates. The first deduplication rate is always for the client data. The second deduplication rate is for the metadata (disk image header and **True Image Restore** information (if applicable)). That information is always deduplicated on a server; typically, deduplication rates for that information are zero or very low.

Additionally, for the client-side deduplication, the first **Info** line displays the **dedupe** and **compression** values separately.

The following job details example excerpt shows the two rates. The **1/8/2013 11:58:09 PM** entry is for the client data; the **1/8/2013 11:58:19 PM** entry is for the metadata.

```
1/8/2013 11:54:21 PM - Info MSDP_Server.example.com(pid=2220)
  Using OpenStorage client direct to backup from client
  Client_B.example.com to MSDP_Server.example.com
1/8/2013 11:58:09 PM - Info MSDP_Server.example.com(pid=2220)
  StorageServer=PureDisk:MSDP_Server.example.com; Report=PDDO
  Stats for (MSDP_Server.example.com: scanned: 110028 KB,
CR sent: 16654 KB, CR sent over FC: 0 KB, dedup: 84.9%,
cache disabled, where dedup space saving:3.8%,
compression space saving:81.1%
1/8/2013 11:58:09 PM - Info MSDP_Server.example.com(pid=2220)
  Using the media server to write NBU data for backup
  Client_B_1254987197.example.com to MSDP_Server.example.com
1/8/2013 11:58:19 PM - Info MSDP_Server.example.com(pid=2220)
  StorageServer=PureDisk:MSDP_Server.example.com; Report=PDDO
  Stats for (MSDP_Server.example.com: scanned: 17161 KB,
CR sent: 17170 KB, dedup: 0.0%, cache hits: 0 (0.0%)
```

Field descriptions

Table 9-1 describes the deduplication activity fields.

Table 9-1 MSDP activity field descriptions

Field	Description
Dedup space saving	The percentage of space that is saved by data deduplication (data is not written again).
Compression space saving	The percentage of space that is saved because the deduplication engine compressed some data before writing it to storage.
cache hits	The percentage of data segments in the backup that is represented in the local fingerprint cache. The deduplication plug-in did not have to query the database about those segments If the `pd.conf` file `FP_CACHE_LOCAL` parameter is set to `0` on the storage, the **cache hits** output is not included for the jobs that run on the storage server. See "MSDP pd.conf file parameters" on page 179.
CR sent	The amount of data that is sent from the deduplication plug-in to the component that stores the data. In NetBackup, the NetBackup Deduplication Engine stores the data. If the storage server deduplicates the data, it does not travel over the network. The deduplicated data travels over the network when the deduplication plug-in runs on a computer other than the storage server, as follows: ■ On a NetBackup client that deduplicates its own data (client-side deduplication). ■ On a fingerprinting media server that deduplicates the data. The deduplication plug-in on the fingerprinting server sends the data to the storage server, which writes it to a **Media Server Deduplication Pool**.

Table 9-1 MSDP activity field descriptions *(continued)*

Field	Description
CR sent over FC	The amount of data that is sent from the deduplication plug-in over Fibre Channel to the component that stores the data. In NetBackup, the NetBackup Deduplication Engine stores the data.
dedup	The percentage of data that was stored already. That data is not stored again.
multi-threaded stream used	Indicates that the Deduplication Multi-Threaded Agent processed the backup. See "About the MSDP Deduplication Multi-Threaded Agent" on page 66.
PDDO stats	Indicates that the job details are for storage on the following destinations: ■ **Media Server Deduplication Pool**
rebased	The percentage of segments that were rebased (that is, defragmented) during the backup. Those segments had poor data locality. NetBackup reports backup job completion only after backup rebasing is completed. See "About MSDP storage rebasing" on page 561.
scanned	The amount of data that the deduplication plug-in scanned.
Using OpenStorage client direct to restore...	Indicates that the restore travels over the client-direct data path and does not use NetBackup media server components to process the data.
encrypted	Indicates if the new transferred data being written to the deduplication pool is encrypted or not.

Descriptions of the job details that are not related to deduplication are in a different topic.

About MSDP storage capacity and usage reporting

Several factors affect the expected NetBackup deduplication capacity and usage results, as follows:

- Expired backups may not change the available size and the used size. An expired backup may have no unique data segments. Therefore, the segments remain valid for other backups.
- NetBackup Deduplication Manager clean-up may not have run yet. The Deduplication Manager performs clean up twice a day. Until it performs clean-up, deleted image fragments remain on disk.

If you use operating system tools to examine storage space usage, their results may differ from the usage reported by NetBackup, as follows:

- NetBackup usage data includes the reserved space that the operating system tools do not include.
- If other applications use the storage, NetBackup cannot report usage accurately. NetBackup requires exclusive use of the storage.

Table 9-2 describes the options for monitoring capacity and usage.

Table 9-2 Capacity and usage reporting

Option	Description
Change Storage Server dialog box	The **Change Storage Server** dialog box **Properties** tab displays storage capacity and usage. It also displays the global deduplication ratio. This dialog box displays the most current capacity usage that is available in the NetBackup web UI. You can see an example of the dialog box in a different topic. See "Monitoring the MSDP deduplication and compression rates" on page 517.
Disk Pools window	The **Disk Pools** window of the NetBackup web UI displays the values that were stored when NetBackup polled the disk pools. NetBackup polls every 5 minutes; therefore, the value may not be as current as the value that is displayed in the **Change Storage Server** dialog box. To display the window **Storage > Disk storage > Disk pools**.
The `crcontrol` command	The `crcontrol` command provides a view of storage capacity and usage within the deduplication container files. See "About MSDP container files" on page 523. See "Viewing storage usage within MSDP container files" on page 523.
Disk Pool Status report	The **Disk Pool Status** report displays the state of the disk pool and usage information.

Table 9-2 Capacity and usage reporting *(continued)*

Option	Description
Disk Logs report	The **Disk Logs** report displays event and message information. A useful event for monitoring capacity is event 1044; the following is the description of the event in the **Disk Logs** report:`The usage of one or more system resources has exceeded a warning level.` By default, the threshold (high-water mark) for this message is at 98% capacity. No more data can be stored. See "MSDP event codes and messages" on page 788.
The `nbdevquery` command	The `nbdevquery` command shows the state of the disk volume and its properties and attributes. It also shows capacity, usage, and percent used. See "Determining the MSDP disk volume state" on page 549.

About MSDP container files

The deduplication storage implementation allocates container files to hold backup data. Deleted segments can leave free space in containers files, but the container file sizes do not change. Segments are deleted from containers when backup images expire and the NetBackup Deduplication Manager performs clean-up.

The NetBackup Deduplication Manager checks the storage space every 20 seconds. It then periodically compacts the space available inside the container files. Therefore, space within a container is not available as soon as it is free. Various internal parameters control whether a container file is compacted. Although space may be available within a container file, the file may not be eligible for compaction.

See "About MSDP storage capacity and usage reporting" on page 521.

See "Viewing storage usage within MSDP container files" on page 523.

See "MSDP storage full conditions" on page 786.

Viewing storage usage within MSDP container files

The NetBackup `crcontrol` command reports on storage usage within containers.

To view storage usage within MSDP container files

- Use the `crcontrol` command and the `--dsstat` option on the deduplication storage server. (For help with the command options, use the `--help` option.)

  The following is an example of the command usage:

  - UNIX and Linux: `/usr/openv/pdde/pdcr/bin/crcontrol --dsstat`
  - `Windows: install_path\\Veritas\\pdde\\Crcontrol.exe --dsstat`

  The following is an example of the output:

```
************ Data Store statistics ************
Data storage   Raw        Size      Used     Avail    Use%  Free%
               199.95GiB  63.70GiB  1.23GiB  62.48GiB  2%   98.1%

Number of containers                : 1
Average container size              : 1049 bytes (0.00MiB)
Space allocated for containers      : 1049 bytes (1.02KiB)
Reserved space                      : 136.25GiB (68.1%)
Reserved space for cloud cache      : 14.00GiB (22.0%)
Reserved space for vpfs cloud cache : 128.00GiB (64.0%)
```

  For systems that host a **Media Server Deduplication Pool**, you can use the following `crcontrol` command to show information about each partition:

```
/usr/openv/pdde/pdcr/bin/crcontrol --dsstat 3
```

  From the command output, you can determine the following:

`Raw`	The raw size of the storage.
`Size`	The size of the storage that NetBackup can use: the `Raw` size of the storage minus the file system `Reserved space`. If the file system has a concept of root reserved space (such as EXT3 or VxFS), that space cannot be used for storage. The `crcontrol` command does not include reserved space in the available space. Unlike the `crcontrol` command, some operating system tools report root reserved space as usable space.
`Used`	The amount of deduplicated data that is stored on the file system. NetBackup obtains the file system used space from the operating system.
`Avail`	The `Size` minus the `Used` space.
`Use%`	The `Used` space divided by the `Size`.

See “About MSDP storage capacity and usage reporting” on page 521.

See “About MSDP container files” on page 523.

See “Processing the MSDP transaction queue manually” on page 553.

See “MSDP storage full conditions” on page 786.

About monitoring MSDP processes

The following table shows the deduplication processes about which NetBackup reports:

See “MSDP server components” on page 593.

Table 9-3 Where to monitor the main MSDP processes

What	Where to monitor it
NetBackup Deduplication Engine	In the NetBackup web UI, the NetBackup Deduplication Engine appears as `spoold` on the **Daemons** tab of the **Activity Monitor**. The NetBackup `bpps` command also shows the `spoold` process.
NetBackup Deduplication Manager	In the NetBackup web UI, the NetBackup Deduplication Manager appears as `spad` on the **Daemons** tab of the **Activity Monitor**. The NetBackup `bpps` command also shows the `spad` process.

See “About MSDP storage servers” on page 35.

See “Introduce MSDP load balancing servers gradually” on page 52.

Reporting on Auto Image Replication jobs

The Activity Monitor displays both the **Replication** job and the **Import** job in a configuration that replicates to a target primary server domain.

Table 9-4 Auto Image Replication jobs in the Activity Monitor

Job type	Description
Replication	The job that replicates a backup image to a target primary displays in the Activity Monitor as a **Replication** job. The **Target Master** label displays in the **Storage Unit** column for this type of job. Similar to other **Replication** jobs, the job that replicates images to a target primary can work on multiple backup images in one instance. The detailed status for this job contains a list of the backup IDs that were replicated.

Table 9-4 Auto Image Replication jobs in the Activity Monitor *(continued)*

Job type	Description
Import	The job that imports a backup copy into the target primary domain displays in the Activity Monitor as an **Import** job. An **Import** job can import multiple copies in one instance. The detailed status for an **Import** job contains a list of processed backup IDs and a list of failed backup IDs. Note that a successful replication does not confirm that the image was imported at the target primary. If the data classifications are not the same in both domains, the **Import** job fails and NetBackup does not attempt to import the image again. Failed **Import** jobs fail with a status 191 and appear in the **Problems** report when run on the target primary server. The image is expired and deleted during an **Image Cleanup** job. Note that the originating domain (Domain 1) does not track failed imports.

Checking the image encryption status

Use `msdpimgutil encryptionreport` command-line utility to check the MSDP pool encryption status or image encryption status on the storage server.

This command-line utility does the following:

- Reports the MSDP pool encryption status.
  It lists the data encryption setting and KMS encryption setting for all LSUs.
- Reports the image data encryption status for the given image.
- Reports the image KMS encryption status for the given image.
- Provides KMS key IDs used for the given encrypted image.
- Provides the unencrypted data container IDs.
- If encryption is disabled, provides encryption protection guidance.

`msdpimgutil encryptionreport` command-line utility is located at the following locations:

- UNIX: `/usr/openv/pdde/pdcr/bin/msdpimgutil`
- Windows: `install_path\Veritas\pdde\msdpimgutil.exe`

To check the image encryption status

1 Run the following command to check the MSDP pool encryption status. It lists data encryption setting and KMS encryption settings for all LSUs.

```
msdpimgutil encryptionreport --systemcheck
```

For example,

```
msdpimgutil encryptionreport --systemcheck
==== Dedup system encryption check ====
Found Local LSU
      LSU's Data encryption setting is enabled.
      LSU's KMS encryption setting is enabled.

Found Cloud LSU aws_vraxmyan9148
      LSU's Data encryption setting is disabled.
      LSU's KMS encryption setting is disabled.

Found Cloud LSU aws2_vraxmyan9148
      LSU's Data encryption setting is enabled.
      LSU's KMS encryption setting is enabled.

** Follow the NetBackup Deduplication Guide to enable Data/KMS
encryption (contentrouter.cfg).

** Encryption Crawler:
   Encryption Crawler is unavailable for WORM Deduplication pools
 or data stored on Cloud Tier.
```

2 Check the encryption settings of the image in the MSDP pool.

It reports the image data encryption status, image KMS encryption status, KMS key IDs, and the unencrypted data container IDs used for the given image. If encryption is disabled, provides encryption protection guidance.

```
msdpimgutil encryptionreport --backupid <backup id> --copy <copy
number> [--verbose | -v]
```

`--backupid`: Backup ID of the image.

`--copy`: Copy number of the image.

`--verbose` or `-v`: Verbose is used to output KMS key IDs for the KMS encrypted image or the data containers IDs for the unencrypted image.

Note: This command may run for a long time if the image consumes a large number of data containers.

For example,

```
msdpimgutil encryptionreport --backupid backupid --copy 1
--verbose
==== Dedup system encryption check ====
Found Local LSU
      LSU's Data encryption setting is enabled.
      LSU's KMS encryption setting is enabled.

==== Dedup image encryption check ====
Found image (2|/client/policy|backupid_C1) in Local LSU.
      Image Data encryption setting is enabled.
      Image KMS encryption setting is enabled.

      Image data segments are found in 16 data containers, 16
data containers are KMS encrypted.

      The following unique KeyID is needed for image recovery.
      KeyID:
87f49487cd13e9d30c4891e146721354f53f26b82945a8b23eb859a0b8416929
```

MSDP events in NetBackup web UI

The NetBackup web UI displays some events that are associated with the MSDP. These events are categorized based on different system components and issues, helping administrators efficiently monitor and troubleshoot.

The following table lists the MSDP events and their categories in the NetBackup web UI.

Table 9-5 MSDP events

Events	Category	Description
MSDP volume group (MVG)	MSDP	Events are triggered by configuration changes, load balancing activities such as workload assignments with physical volumes, and when physical volumes go offline or are marked as full.
CRC check	Spoold cyclic redundancy check	Events are generated for data corruption issues, CRC check progress, or if CRC checks have not been run for a long period. See "About the CRC notifications for local LSU" on page 559.
Universal share health	MSDP	Universal share health status is monitored for conditions such as a healthy state, mount point hangs, or insufficient disk space.
Universal share (VPFS) alerts	VPFS	Specific alerts are raised when universal share has not had a backup or checkpoint for an extended period, or when corruption is detected in files or data within a group.

Chapter 10

Managing deduplication

This chapter includes the following topics:

- Managing MSDP servers
- Managing NetBackup Deduplication Engine credentials
- Managing Media Server Deduplication Pools
- Analyzing the disc space consumption of the backup images
- Deleting backup images
- About MSDP queue processing
- Processing the MSDP transaction queue manually
- About MSDP data integrity checking
- About managing MSDP storage read performance
- About MSDP storage rebasing
- About the MSDP data removal process
- Resizing the MSDP storage partition
- How MSDP restores work
- Configuring MSDP restores directly to a client
- About restoring files at a remote site
- About restoring from a backup at a target primary domain
- Specifying the restore server
- Enabling extra OS STIG hardening on WORM storage server instance

- Using multiple MSDP nodes for multistream backups on MSDP cluster
- Enabling the media server and MSDP engine affinity in the MSDP cluster

Managing MSDP servers

After you configure deduplication, you can perform various tasks to manage deduplication servers.

See "Viewing MSDP storage servers" on page 531.

See "Determining the MSDP storage server state" on page 531.

See "Viewing MSDP storage server attributes" on page 532.

See "Setting MSDP storage server attributes" on page 533.

See "Changing MSDP storage server properties" on page 534.

See "Clearing MSDP storage server attributes" on page 534.

See "About changing the MSDP storage server name or storage path" on page 535.

See "Changing the MSDP storage server name or storage path" on page 536.

See "Removing an MSDP load balancing server" on page 537.

See "Deleting an MSDP storage server" on page 538.

See "Deleting the MSDP storage server configuration" on page 539.

Viewing MSDP storage servers

You can view a list of deduplication storage servers that are configured in NetBackup.

To view MSDP storage servers

1. Open the web UI.
2. On the left, click **Storage > Disk storage**.
3. Click the **Storage servers** tab.

   This list shows all configured storage servers. Deduplication storage servers show **PureDisk** in the **Storage server type** column.

Determining the MSDP storage server state

Use the NetBackup `nbdevquery` command to determine the state of a deduplication storage server. The state is either UP or DOWN.

To determine MSDP storage server state

- Run the following command on the NetBackup primary server or a deduplication storage server:

  UNIX: `/usr/openv/netbackup/bin/admincmd/nbdevquery -liststs -storage_server server_name -stype PureDisk –U`

  Windows: `install_path\NetBackup\bin\admincmd\nbdevquery -liststs -storage_server server_name -stype PureDisk –U`

  The following is example output:

  ```
  Storage Server      : bit.example.com
  Storage Server Type : PureDisk
  Storage Type        : Formatted Disk, Network Attached
  State               : UP
  ```

  This example output is shortened; more flags may appear in actual output.

Viewing MSDP storage server attributes

Use the NetBackup `nbdevquery` command to view the deduplication storage server attributes.

The *server_name* you use in the `nbdevquery` command must match the configured name of the storage server. If the storage server name is its fully-qualified domain name, you must use that for *server_name*.

To view MSDP storage server attributes

- The following is the command syntax to set a storage server attribute. Run the command on the NetBackup primary server or on the deduplication storage server:

  UNIX: `/usr/openv/netbackup/bin/admincmd/nbdevquery -liststs -storage_server server_name -stype PureDisk -U`

  Windows: `install_path\NetBackup\bin\admincmd\nbdevquery -liststs -storage_server server_name -stype PureDisk -U`

  The following is example output:

```
Storage Server      : bit
Storage Server Type : PureDisk
Storage Type        : Formatted Disk, Network Attached
State               : UP
Flag                : OpenStorage
Flag                : CopyExtents
Flag                : AdminUp
Flag                : InternalUp
Flag                : LifeCycle
Flag                : CapacityMgmt
Flag                : OptimizedImage
Flag                : FT-Transfer
```

  This example output is shortened; more flags may appear in actual output.

Setting MSDP storage server attributes

You may have to set storage server attributes to enable new functionality.

If you set an attribute on the storage server, you may have to set the same attribute on existing deduplication pools. The overview or configuration procedure for the new functionality describes the requirements.

See "Setting a Media Server Deduplication Pool attribute" on page 543.

To set a MSDP storage server attribute

1. The following is the command syntax to set a storage server attribute. Run the command on the primary server or on the storage server.

   `nbdevconfig -changests -storage_server storage_server -stype PureDisk -setattribute attribute`

   The following describes the options that require the arguments that are specific to your domain:

`-storage_server` *storage_server*	The name of the storage server.
`-setattribute` *attribute*	The *attribute* is the name of the argument that represents the new functionality. For example, **OptimizedImage** specifies that the environment supports the optimized synthetic backup method.

The following is the path to the `nbdevconfig` command:

- UNIX: /usr/openv/netbackup/bin/admincmd
- Windows: *install_path*\NetBackup\bin\admincmd

2 To verify, view the storage server attributes.

See “Viewing MSDP storage server attributes” on page 532.

See “About MSDP optimized synthetic backups” on page 44.

Changing MSDP storage server properties

You can change the retention period and logging level for the NetBackup Deduplication Manager.

To change MSDP storage server properties

1 Open the web UI.

2 On the left, click **Storage > Disk storage**.

3 Click the **Storage servers** tab.

4 Click on name of the deduplication storage server.

5 Click **Edit** and make the wanted changes.

6 Click **Save**.

See “About MSDP deduplication nodes” on page 27.

See “About MSDP storage servers” on page 35.

See “MSDP storage path properties” on page 95.

See “Configuring a storage server for a Media Server Deduplication Pool” on page 93.

Clearing MSDP storage server attributes

Use the `nbdevconfig` command to remove storage server attributes.

To clear MSDP storage server attributes

- Run the following command on the NetBackup primary server or on a storage server:

  ```
  /usr/openv/netbackup/bin/admincmd/nbdevconfig -changests
  -storage_server storage_server -stype PureDisk -clearattribute
  attribute
  ```

  ```
  nbdevconfig -changests -storage_server storage_server -stype
  PureDisk -clearattribute attribute
  ```

`-storage_server` *`storage_server`*	The name of the storage server.
`-setattribute` *`attribute`*	The *attribute* is the name of the argument that represents the functionality.

  The following is the path to the `nbdevconfig` command:

  - UNIX: `/usr/openv/netbackup/bin/admincmd`
  - Windows: *`install_path`*`\NetBackup\bin\admincmd`

About changing the MSDP storage server name or storage path

You can change the storage server host name and the storage path of an existing NetBackup deduplication environment.

The following are several use cases that require changing an existing deduplication environment:

- You want to change the host name. For example, the name of host A was changed to B or a new network card was installed with a private interface C. To use the host name B or the private interface C, you must reconfigure the storage server.
  See "Changing the MSDP storage server name or storage path" on page 536.
- You want to change the storage path. To do so, you must reconfigure the storage server with the new path.
  See "Changing the MSDP storage server name or storage path" on page 536.
- You need to reuse the storage for disaster recovery. The storage is intact, but the storage server was destroyed. To recover, you must configure a new storage server.
  In this scenario, you can use the same host name and storage path or use different ones.
  See "Recovering from an MSDP storage server failure" on page 584.

Changing the MSDP storage server name or storage path

Two aspects of a NetBackup deduplication configuration exist: the record of the deduplication storage in the EMM database and the physical presence of the storage on disk (the populated storage directory).

Warning: Deleting valid backup images may cause data loss.

See "About changing the MSDP storage server name or storage path" on page 535.

Table 10-1 Changing the storage server name or storage path

Step	Task	Procedure
Step 1	Ensure that no deduplication activity occurs	Deactivate all backup policies that use deduplication storage. See theNetBackup Administrator's Guide, Volume I.
Step 2	Expire the backup images	Expire all backup images that reside on the deduplication disk storage. **Warning:** Do not delete the images. They are imported back into NetBackup later in this process. If you use the `bpexpdate` command to expire the backup images, use the `-nodelete` parameter. See the*NetBackup Administrator's Guide, Volume I*.
Step 3	Delete the storage units that use the disk pool	See the*NetBackup Administrator's Guide, Volume I*.
Step 4	Delete the disk pool	See "Deleting a Media Server Deduplication Pool" on page 550.
Step 5	Delete the deduplication storage server	See "Deleting an MSDP storage server" on page 538.
Step 6	Delete the configuration	Delete the deduplication configuration. See "Deleting the MSDP storage server configuration" on page 539.
Step 7	Delete the deduplication host configuration file	Each load balancing server contains a deduplication host configuration file. If you use load balancing servers, delete the deduplication host configuration file from those servers. See "Deleting an MSDP host configuration file" on page 200.

Table 10-1 Changing the storage server name or storage path *(continued)*

Step	Task	Procedure
Step 8	Delete the identity file and the file system table file	Delete the following files from the MSDP storage server, depending on operating system: UNIX: `/storage_path/data/.identity` `/storage_path/etc/puredisk/fstab.cfg` Windows: `storage_path\data\.identity` `storage_path\etc\puredisk\fstab.cfg`
Step 9	Change the storage server name or the storage location	See the computer or the storage vendor's documentation. See "Use fully qualified domain names" on page 51. See "MSDP storage path properties" on page 95.
Step 10	Reconfigure the storage server	When you configure deduplication, select the host by the new name and enter the new storage path (if you changed the path). You can also use a new network interface. See "Configuring media server deduplication in NetBackup" on page 63.
Step 11	Import the backup images	See theNetBackup Administrator's Guide, Volume I.

Removing an MSDP load balancing server

You can remove a load balancing server from a deduplication node. The media server no longer deduplicates client data.

See "About MSDP storage servers" on page 35.

After you remove the load balancing server, restart the NetBackup Enterprise Media Manager service. The NetBackup disk polling service may try to use the removed server to query for disk status. Because the server is no longer a load balancing server, it cannot query the disk storage. Consequently, NetBackup may mark the disk volume as DOWN. When the EMM service restarts, it chooses a different deduplication server to monitor the disk storage.

If the host failed and is unavailable, you can use the `tpconfig` device configuration utility in menu mode to delete the server. However, you must run the `tpconfig` utility on a UNIX or Linux NetBackup server.

For procedures, see the NetBackup Administrator's Guide, Volume II.

To remove a media server from an MSDP node

1. Open the web UI.
2. On the left, click **Storage > Disk storage**.
3. Click the **Storage servers** tab.
4. Review the **Media servers** column. For every storage unit where you manually selected one or more media servers, select the option **Allow NetBackup to automatically select** (Any available).
5. Click the storage server name.
6. In the **Media servers** list, locate the media server and click **Delete** to remove the media server from an MSDP node.

See "About MSDP server requirements" on page 37.

See "About scaling MSDP" on page 51.

See "Introduce MSDP load balancing servers gradually" on page 52.

Deleting an MSDP storage server

If you delete a deduplication storage server, NetBackup deletes the host as a storage server and disables the deduplication storage server functionality on that media server.

NetBackup does not delete the media server from your configuration. To delete the media server, use the NetBackup `nbemmcmd` command.

Deleting the deduplication storage server does not alter the contents of the storage on physical disk. To protect against inadvertent data loss, NetBackup does not automatically delete the storage when you delete the storage server.

If a disk pool is configured from the disk volume that the deduplication storage server manages, you cannot delete the deduplication storage server.

Warning: Do not delete a deduplication storage server if its storage contains unexpired NetBackup images; if you do, data loss may occur.

To delete an MSDP storage server

1. Open the web UI.
2. On the left, click **Storage > Disk storage**.
3. Click the **Storage servers** tab.
4. Select the storage server and click **Delete**.

See "Changing the MSDP storage server name or storage path" on page 536.

Deleting the MSDP storage server configuration

Use this procedure to delete a deduplication storage server configuration. The script that is used in this procedure deletes the active configuration and returns the configuration files to their installed, preconfigured state.

Only use this procedure when directed to from a process topic. A process topic is a high-level user task made up of a series of separate procedures.

See "Changing the MSDP storage server name or storage path" on page 536.

See "Deactivating MSDP" on page 591.

To delete the MSDP storage server configuration

1. Use the **NetBackup web UI** to stop the NetBackup Deduplication Engine (`spoold`) and the NetBackup Deduplication Manager (`spad`).
2. On the storage server, run one of the following scripts, depending on your operating system:

   UNIX:
   `/usr/openv/pdde/pdconfigure/scripts/installers/PDDE_deleteConfig.sh`

   Windows: `install_path\Program Files\Veritas\pdde\PDDE_deleteConfig.bat`

   The command output includes the following:

   ```
   **** Starting PDDE_deleteConfig.sh ***
   You need to stop the spad and spoold daemons to proceed
   This script will delete the PDDE configuration on this system
   Would you want to continue? [ y | n ]
   ```

3. Type **y** and then press Enter.

Managing NetBackup Deduplication Engine credentials

You can manage existing credentials in NetBackup.

See "Determining which media servers have deduplication credentials" on page 540.

See "Adding NetBackup Deduplication Engine credentials" on page 540.

See "Changing NetBackup Deduplication Engine credentials" on page 541.

See "Deleting credentials from a load balancing server" on page 541.

Determining which media servers have deduplication credentials

You can determine which media servers have credentials configured for the NetBackup Deduplication Engine. The servers with credentials are load balancing servers.

To determine if NetBackup Deduplication Engine credentials exist

1. Open the web UI.
2. On the left, click **Storage > Disk storage**.
3. Click the **Storage servers** tab.
4. Click on the storage server name.
5. Locate the **Media servers** list.

Adding NetBackup Deduplication Engine credentials

You may need to add the NetBackup Deduplication Engine credentials to an existing storage server or load balancing server. For example, disaster recovery may require that you add the credentials.

Add the same credentials that you already use in your environment.

Another procedure exists to add a load balancing server to your configuration.

See "Adding an MSDP load balancing server" on page 174.

To add NetBackup Deduplication Engine credentials by using the tpconfig command

- On the host to which you want to add credentials, run the following command:

  On Windows:

  ```
  install_path\Veritas\NetBackup\Volmgr\bin\tpconfig -add
  -storage_server sshostname -stype PureDisk -sts_user_id UserID
  -password PassWord
  ```

  On UNIX/Linux:

  ```
  /usr/openv/volmgr/bin/tpconfig -add -storage_server sshostname
  -stype PureDisk -sts_user_id UserID -password PassWord
  ```

  For *sshostname*, use the name of the storage server.

Changing NetBackup Deduplication Engine credentials

You cannot change the NetBackup Deduplication Engine credentials after you enter them. If you must change the credentials, contact your Cohesity support representative.

See "About the NetBackup Deduplication Engine credentials" on page 41.

Deleting credentials from a load balancing server

You may need to delete the NetBackup Deduplication Engine credentials from a load balancing server. For example, disaster recovery may require that you delete the credentials on a load balancing server.

Another procedure exists to remove a load balancing server from a deduplication node.

See "Removing an MSDP load balancing server" on page 537.

To delete credentials from a load balancing server

- On the load balancing server, run the following command:

  On Windows:

  ```
  install_path\Veritas\NetBackup\Volmgr\bin\tpconfig -delete
  -storage_server sshostname -stype PureDisk -sts_user_id UserID
  ```

  On UNIX/Linux:

  ```
  /usr/openv/volmgr/bin/tpconfig -delete -storage_server sshostname
  -stype PureDisk -sts_user_id UserID
  ```

  For *sshostname*, use the name of the storage server.

Managing Media Server Deduplication Pools

After you configure NetBackup deduplication, you can perform various tasks to manage your deduplication disk pools.

See "Viewing Media Server Deduplication Pools" on page 542.

See "Changing a Media Server Deduplication Pool properties" on page 544.

See "Determining the Media Server Deduplication Pool state" on page 542.

See "Determining the MSDP disk volume state" on page 549.

See "Changing the MSDP disk volume state" on page 550.

See "Viewing Media Server Deduplication Pool attributes" on page 542.

See “Setting a Media Server Deduplication Pool attribute” on page 543.

See “Clearing a Media Server Deduplication Pool attribute” on page 548.

See “Resizing the MSDP storage partition” on page 564.

See “Deleting a Media Server Deduplication Pool” on page 550.

Viewing Media Server Deduplication Pools

You can view the configured disk pools.

To view disk pools

1. Open the NetBackup web UI.
2. On the left, click **Storage > Disk storage**.
3. Click the **Disk pools** tab.

Determining the Media Server Deduplication Pool state

The disk pool state is UP or DOWN.

To determine disk pool state

1. Open the NetBackup web UI.
2. On the left, click **Storage > Disk storage**.
3. Click the **Disk pools** tab.
4. Review the **Status** column.

Viewing Media Server Deduplication Pool attributes

Use the NetBackup `nbdevquery` command to view deduplication pool attributes.

To view MSDP pool attributes

- The following is the command syntax to view the attributes of a deduplication pool. Run the command on the NetBackup primary server or on the deduplication storage server:

  UNIX: `/usr/openv/netbackup/bin/admincmd/nbdevquery -listdp -dp pool_name -stype PureDisk –U`

  Windows: `install_path\NetBackup\bin\admincmd\nbdevquery -listdp -dp pool_name -stype PureDisk –U`

  The following is example output:

```
Disk Pool Name   : MediaServerDeduplicationPool
Disk Pool Id     : MediaServerDeduplicationPool
Disk Type        : PureDisk
Status           : UP
Flag             : OpenStorage
Flag             : AdminUp
Flag             : InternalUp
Flag             : LifeCycle
Flag             : CapacityMgmt
Flag             : OptimizedImage
Raw Size (GB)    : 235.76
Usable Size (GB) : 235.76
Num Volumes      : 1
High Watermark   : 98
Low Watermark    : 80
Max IO Streams   : -1
Storage Server   : DedupeServer.example.com (UP)
```

  This example output is shortened; more flags may appear in actual output.

Setting a Media Server Deduplication Pool attribute

You may have to set attributes on your existing media server deduplication pools. For example, if you set an attribute on the storage server, you may have to set the same attribute on your existing deduplication disk pools.

To set a MSDP disk pool attribute

1. The following is the command syntax to set a deduplication pool attribute. Run the command on the primary server or on the storage server.

   `nbdevconfig -changedp -dp pool_name -stype PureDisk -setattribute attribute`

The following describes the options that require the arguments that are specific to your domain:

`-changedp` *`pool_name`*	The name of the disk pool.
`-setattribute` *`attribute`*	The *attribute* is the name of the argument that represents the new functionality. For example, **OptimizedImage** specifies that the environment supports the optimized synthetic backup method.

The following is the path to the `nbdevconfig` command:

- UNIX: `/usr/openv/netbackup/bin/admincmd`
- Windows: *`install_path`*`\NetBackup\bin\admincmd`

2 To verify, view the disk pool attributes.

See "Viewing Media Server Deduplication Pool attributes" on page 542.

See "About disk pools for NetBackup deduplication" on page 97.

Changing a Media Server Deduplication Pool properties

You can change the properties of a deduplication disk pool.

To change disk pool properties

1 Open the NetBackup web UI.

2 On the left, click **Storage > Disk storage**.

3 Click the **Disk pools** tab.

4 Click the name of the disk pool.

5 Click the **Details** tab.

6 Click **Edit** and make the wanted changes.

7 Click **Save**.

How to resolve volume changes for Auto Image Replication

When you edit a disk pool, NetBackup loads the disk pool properties from the catalog. NetBackup queries the storage server for changes when you either click the **Refresh** button on the **Disk pools** tab or when you select the **Update disk volume** option.

It is recommended that you take the following actions when the volume topology changes:

- Discuss the changes with the storage administrator. You need to understand the changes so you can change your disk pools (if required) so that NetBackup can continue to use them.
- If the changes were not planned for NetBackup, ask your storage administrator to revert the changes so that NetBackup functions correctly again.

NetBackup can process changes to the following volume properties:

- Replication source
- Replication target
- None

If these volume properties change, NetBackup can update the disk pool to match the changes. NetBackup can continue to use the disk pool, although the disk pool may no longer match the storage unit or storage lifecycle purpose.

The following table describes the possible outcomes and how to resolve them.

Table 10-2 Refresh outcomes

Outcome	Description
No changes are discovered.	No changes are required.
NetBackup discovers the new volumes that you can add to the disk pool.	The new volumes display in the properties for that Disk pool.

Table 10-2 Refresh outcomes *(continued)*

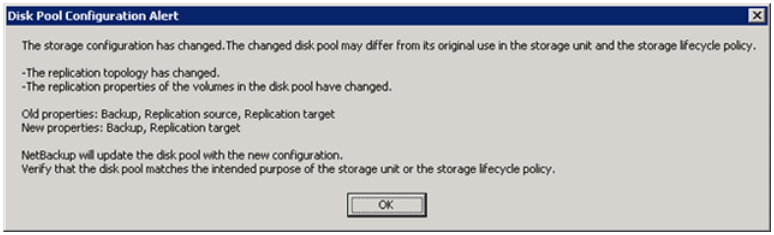
Outcome	Description
The replication properties of all of the volumes changed, but they are still consistent.	A **Disk Pool Configuration Alert** pop-up box notifies you that the properties of all of the volumes in the disk pool changed, but they are all the same (homogeneous).  You must click **OK** in the alert box, after which the disk pool properties in the **Change Disk Pool** dialog box are updated to match the new volume properties If new volumes are available that match the new properties, NetBackup displays those volumes in the **Change Disk Pool** dialog box. You can add those new volumes to the disk pool. In the **Change Disk Pool** dialog box, select one of the following two choices: ■ **OK**. To accept the disk pool changes, click **OK** in the **Change Disk Pool** dialog box. NetBackup saves the new properties of the disk pool. NetBackup can use the disk pool, but it may no longer match the intended purpose of the storage unit or storage lifecycle policy. Change the storage lifecycle policy definitions to ensure that the replication operations use the correct source and target disk pools, storage units, and storage unit groups. Alternatively, work with your storage administrator to change the volume properties back to their original values. ■ **Cancel**. To discard the changes, click **Cancel** in the **Change Disk Pool** dialog box. NetBackup does not save the new disk pool properties. NetBackup can use the disk pool, but it may no longer match the intended use of the storage unit or storage lifecycle policy.

Table 10-2 Refresh outcomes *(continued)*

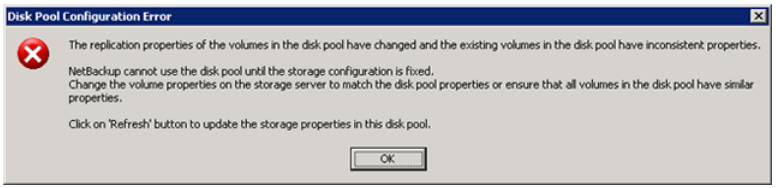
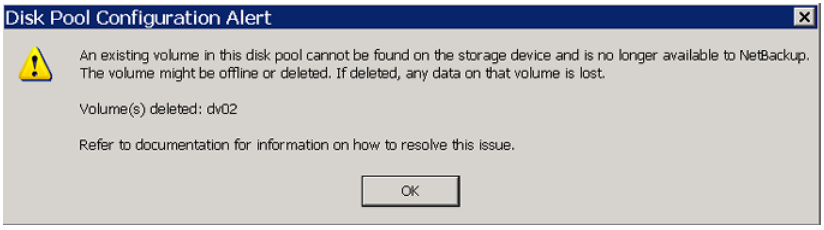
Outcome	Description
The replication properties of the volumes changed, and they are now inconsistent.	A **Disk Pool Configuration Error** pop-up box notifies you that the replication properties of some of the volumes in the disk pool changed. The properties of the volumes in the disk pool are not homogeneous.  You must click **OK** in the alert box. In the **Change Disk Pool** dialog box, the properties of the disk pool are unchanged, and you cannot select them (that is, they are dimmed). However, the properties of the individual volumes are updated. Because the volume properties are not homogeneous, NetBackup cannot use the disk pool until the storage configuration is fixed. NetBackup does not display new volumes (if available) because the volumes already in the disk pool are not homogeneous. To determine what has changed, compare the disk pool properties to the volume properties. See "Viewing the replication topology for Auto Image Replication" on page 144. Work with your storage administrator to understand the changes and why they were made. The replication relationships may or may not have to be re-established. If the relationship was removed in error, re-establishing the relationships seem justified. If you are retiring or replacing the target replication device, you probably do not want to re-establish the relationships. The disk pool remains unusable until the properties of the volumes in the disk pool are homogenous. In the **Change Disk Pool** dialog box, click **OK** or **Cancel** to exit the **Change Disk Pool** dialog box.

Table 10-2 Refresh outcomes *(continued)*

Outcome	Description
NetBackup cannot find a volume or volumes that were in the disk pool.	A **Disk Pool Configuration Alert** pop-up box notifies you that an existing volume or volumes was deleted from the storage device:  NetBackup can use the disk pool, but data may be lost. To protect against accidental data loss, NetBackup does not allow volumes to be deleted from a disk pool. To continue to use the disk pool, do the following: ■ Use the `bpimmedia` command or the Images on Disk report to display the images on the specific volume. ■ Expire the images on the volume. ■ Use the `nbdevconfig` command to set the volume state to DOWN so NetBackup does not try to use it.

Clearing a Media Server Deduplication Pool attribute

You may have to clear attributes on your existing media server deduplication pools.

To clear a Media Server Deduplication Pool attribute

◆ The following is the command syntax to clear a deduplication pool attribute. Run the command on the primary server or on the storage server.

```
nbdevconfig -changedp -dp pool_name -stype PureDisk
-clearattribute attribute
```

The following describe the options that require your input:

`-changedp` *`pool_name`*	The name of the disk pool.
`-setattribute` *`attribute`*	The *attribute* is the name of the argument that represents the new functionality.

The following is the path to the `nbdevconfig` command:

- UNIX: `/usr/openv/netbackup/bin/admincmd`
- Windows: *`install_path`*`\NetBackup\bin\admincmd`

Determining the MSDP disk volume state

Use the NetBackup `nbdevquery` command to determine the state of the volume in a deduplication disk pool. NetBackup exposes all of the storage for MSDP as a single volume, **PureDiskVolume**. The command shows the properties and attributes of the **PureDiskVolume**.

To determine MSDP disk volume state

- Display the volume state by using the following command:

  UNIX: `/usr/openv/netbackup/bin/admincmd/nbdevquery -listdv -stype PureDisk -U -dp` *`disk_pool_name`*

  Windows: *`install_path`*`\NetBackup\bin\admincmd\nbdevquery -listdv -stype PureDisk -U -dp` *`disk_pool_name`*

  The *state* is either UP or DOWN.

  The following is example output

```
Disk Pool Name      : MSDP_Disk_Pool
Disk Type           : PureDisk
Disk Volume Name    : PureDiskVolume
Disk Media ID       : @aaaab
Total Capacity (GB) : 49.98
Free Space (GB)     : 43.66
Use%                : 12
Status              : UP
Flag                : ReadOnWrite
Flag                : AdminUp
Flag                : InternalUp
Num Read Mounts     : 0
Num Write Mounts    : 1
Cur Read Streams    : 0
Cur Write Streams   : 0
```

See "Changing the MSDP disk volume state" on page 550.

See "About the MSDP pd.conf configuration file" on page 178.

See "Recovering the MSDP storage server after NetBackup catalog recovery" on page 587.

Changing the MSDP disk volume state

The disk volume state is **UP** or **DOWN**. NetBackup exposes all of the storage for MSDP as a single volume, **PureDiskVolume**.

To change the state to **DOWN**, the disk pool in which the volume resides must not be busy. If backup jobs are assigned to the disk pool, the state change fails. Cancel the backup jobs or wait until the jobs complete.

To change the MSDP disk volume state

- Change the disk volume state; the following is the command syntax:

  UNIX: `/usr/openv/netbackup/bin/admincmd/nbdevconfig -changestate -stype PureDisk -dp disk_pool_name -dv PureDiskVolume -state state`

  Windows: `install_path\NetBackup\bin\admincmd\nbdevconfig -changestate -stype PureDisk -dp disk_pool_name -dv PureDiskVolume -state state`

  For the `-state`, specify either **UP** or **DOWN**.

See "Determining the MSDP disk volume state" on page 549.

See "About the MSDP pd.conf configuration file" on page 178.

See "Recovering the MSDP storage server after NetBackup catalog recovery" on page 587.

Deleting a Media Server Deduplication Pool

You can delete a disk pool if it does not contain valid NetBackup backup images or image fragments. If it does, you must first expire and delete those images or fragments. If expired image fragments remain on disk, you must remove those also.

See "Cannot delete an MSDP disk pool" on page 781.

If you delete a disk pool, NetBackup removes it from your configuration.

If a disk pool is the storage destination of a storage unit, you must first delete the storage unit.

To delete an MSDP disk pool

1. Open the NetBackup web UI.
2. On the left, click **Storage > Disk storage**.
3. Select a disk pool.
4. Click **Delete > Yes**.

Analyzing the disc space consumption of the backup images

Use `msdpimgutil spacereport` command to analyze how much disk space is consumed by the backup images. This utility helps you identify images with unexpected deduplication rates and make a right decision on what images to expire in the event of high deduplication pool space usage.

The `msdpimgutil` utility is located at the following location:

- UNIX: `/usr/openv/pdde/pdcr/bin/msdpimgutil`
- Windows: `install_path\Veritas\pdde\msdpimgutil.exe`

Limitations:

- The command does not support universal share dump data statistics.
- When you run this command to check the data size, do not expire the backup images that you specify in the command input parameters.
- Ensure that the expired storage space is reclaimed before you run this utility.
- The command does not support the **MSDP-Object-Store** policy type data statistics.

Run the following command with the required options to analyze the disk space consumed.

```
msdpimgutil spacereport <[--backup_id_list <backupid list>] [--client
<client name>] [--policy <policy name>] [--startdate <start date>]
[--enddate <end date>]> [--copynumber <copy number>]
```

For example,

```
msdpimgutil spacereport --client sadiexxvmxx.xxx.xxx.veritas.com
--policy PCloud --startdate 2023-09-02T01:23:22 --enddate 2023-
12-03T01:23:22 -copynumber 1
```

To analyze the disk space consumption of the backup images

1. Get the size consumed by the specified backup images.

   ```
   msdpimgutil spacereport --backup_id_list
   sadiexxvmxx.xxx.xxx.veritas.com_xxxxx02360
   sadiexxvmxx.xxx.xxx.veritas.com_xxxxx02243
   ```

2. Get the size consumed by all the backup images from a specified client.

   ```
   msdpimgutil spacereport --client sadiexxvmxx.xxx.xxx.veritas.com
   ```

3 Get the size consumed by all backup images from a specified policy or client and policy.

```
msdpimgutil spacereport --client sadiexxvmxx.xxx.xxx.veritas.com
--policy dirPlocal2
```

4 Get the size consumed by all the backup images between the specified start date and end date.

```
msdpimgutil spacereport --startdate 2023-09-02T01:23:22 --enddate
2023-12-03T01:23:22
```

Deleting backup images

Image deletion may be time consuming. Therefore, if you delete images manually, Cohesity recommends the following approach.

See "About the MSDP data removal process" on page 563.

To delete backup images manually

1 Expire all of the images by using the `bpexpdate` command and the `-notimmediate` option. The `-notimmediate` option prevents `bpexpdate` from calling the `nbdelete` command, which deletes the image.

Without this option, `bpexpdate` calls `nbdelete` to delete images. Each call to `nbdelete` creates a job in the Activity Monitor, allocates resources, and launches processes on the media server.

2 After you expire the last image, delete all of the images by using the `nbdelete` command with the `-allvolumes` option.

Only one job is created in the Activity Monitor, fewer resources are allocated, and fewer processes are started on the media servers. The entire process of expiring images and deleting images takes less time.

About MSDP queue processing

Operations that require database updates accumulate in a transaction queue. Twice a day, the NetBackup Deduplication Manager directs the Deduplication Engine to process the queue as one batch. By default, queue processing occurs every 12 hours, 20 minutes past the hour.

Primarily, the transaction queue contains clean-up and integrity checking transactions. These transactions update the reference database.

Queue processing writes status information to the deduplication engine `storaged.log` file.

See “NetBackup MSDP log files” on page 767.

Because queue processing does not block any other deduplication process, rescheduling should not be necessary. Users cannot change the maintenance process schedules. However, if you must reschedule these processes, contact your Cohesity support representative.

Because queue processing occurs automatically, you should not need to invoke it manually. However, you may do so.

See “Processing the MSDP transaction queue manually” on page 553.

See “About MSDP server requirements” on page 37.

Processing the MSDP transaction queue manually

NetBackup maintains a queue for MSDP database transactions.

See “About MSDP queue processing” on page 552.

Usually, you should not need to run the deduplication database transaction queue processes manually. However, when you recover the MSDP catalog from a backup, you must process the MSDP transaction queue. Processing the transaction queue is part of a larger process.

By default, MSDP processs all Local and Cloud LSU database transaction queue. However, you can run queue processes by cloud LSU or local LSU individually by providing a cloud LSU dsid value. Use `/usr/openv/pdde/pdcr/bin/pddecfg -a listcloudlsu` to get cloud LSU dsid value. If given dsid value is “0”, local LSU is processed.

To process the MSDP transaction queue manually

1. On the MSDP storage server, run the following command:

   UNIX: `/usr/openv/pdde/pdcr/bin/crcontrol --processqueue --dsid <dsid>`

   Windows: *install_path*`\Veritas\pdde\Crcontrol.exe --processqueue --dsid <dsid>`

   `--dsid` is the optional parameter. Without dsid value, all local and cloud LSU process the MSDP transaction queue.

2. To determine if the queue processing is still active, run the following command:

   UNIX: `/usr/openv/pdde/pdcr/bin/crcontrol --processqueueinfo --dsid <dsid>`

   Windows: *install_path*`\Veritas\pdde\Crcontrol.exe --processqueueinfo --dsid <dsid>`

   If the output shows `Busy : yes`, the queue is still active.

   `--dsid` is optional parameter. Without dsid value, if any of local or cloud LSU is active, the command output is `busy`.

3. To examine the results, run the following command (number 1 not lowercase letter l):

   UNIX: `/usr/openv/pdde/pdcr/bin/crcontrol --dsstat 1`

   Windows: *install_path*`\Veritas\pdde\Crcontrol.exe --dsstat 1`

   The command may run for a long time; if you omit the `1`, results return more quickly but they are not as accurate.

About MSDP data integrity checking

Deduplication metadata and data may become inconsistent or corrupted because of disk failures, I/O errors, database corruption, and operational errors. NetBackup checks the integrity of the deduplicated data on a regular basis. NetBackup performs some of the integrity checking when the storage server is idle. Other integrity checking is designed to use few storage server resources so as not to interfere with operations.

The data integrity checking process includes the following checks and actions:

- Automatically constrains data loss or corruption to ensure that new backups are intact.
- Automatically runs a cyclic redundancy check (CRC) for the data containers.

- Automatically collects and cleans up storage garbage.
- Automatically recovers the container-based reference database (or parts of the database) if it is corrupt or missing.
- Automatically finds storage leaks and fixes them.

NetBackup resolves many integrity issues without user intervention, and some issues are fixed when the next backup runs. However, a severe issue may require intervention by Cohesity Support. In such cases, NetBackup writes a message to the NetBackup Disk Logs report.

The data integrity message code is 1057.

See "MSDP event codes and messages" on page 788.

NetBackup writes the integrity checking activity messages to the NetBackup Deduplication Engine `storaged.log` file. For cloud LSU, the messages were written to`Storaged_<dsid>.log`.

See "NetBackup MSDP log files" on page 767.

You can configure some of the data integrity checking behaviors.

See "Configuring MSDP data integrity checking behavior" on page 555.

Configuring MSDP data integrity checking behavior

NetBackup performs several data integrity checks. You can configure the behavior of the integrity checks. For cloud LSU, you can configure the behavior individually for different cloud LSU by the `dsid` value.

Two methods exist to configure MSDP data integrity checking behavior, as follows:

- Run a command.
  See "To configure data integrity checking behavior by using a command" on page 556.
- Edit configuration file parameters.
  See "To configure data integrity checking behavior by editing the configuration files" on page 557.

Warning: Cohesity recommends that you do not disable the data integrity checking. If you do so, NetBackup cannot find and repair or report data corruption.

See "About MSDP data integrity checking" on page 554.

See "MSDP data integrity checking configuration parameters" on page 557.

To configure data integrity checking behavior by using a command

- To configure behavior, specify a value for each of the data integrity checks, as follows:
  - Data consistency checking. Use the following commands to configure behavior:

Enable	UNIX: `/usr/openv/pdde/pdcr/bin/pddecfg -a enabledataintegritycheck -d <dsid>` Windows: *install_path*`\Veritas\pdde\pddecfg -a enabledataintegritycheck -d <dsid>`
Disable	UNIX: `/usr/openv/pdde/pdcr/bin/pddecfg -a disabledataintegritycheck -d <dsid>` Windows: *install_path*`\Veritas\pdde\pddecfg -a disabledataintegritycheck -d <dsid>`
Get the status	UNIX: `/usr/openv/pdde/pdcr/bin/pddecfg -a getdataintegritycheck -d <dsid>` Windows: *install_path*`\Veritas\pdde\pddecfg -a getdataintegritycheck -d <dsid>`

Note: `-d` is cloud LSU dsid value and it is an optional parameter. Use `/usr/openv/pdde/pdcr/bin/pddecfg -a listcloudlsu` to get cloud LSU dsid value. When the dsid value is “0”, local LSU is processed.

  - Cyclic redundancy checking. Use the following commands to configure behavior:

Enable	CRC does not run if queue processing is active or during disk read or write operations. UNIX: `/usr/openv/pdde/pdcr/bin/crcontrol --crccheckon` Windows: *install_path*`\Veritas\pdde\Crcontrol.exe --crccheckon`
Disable	UNIX: `/usr/openv/pdde/pdcr/bin/crcontrol --crccheckoff` Windows: *install_path*`\Veritas\pdde\Crcontrol.exe --crccheckoff`

Enable fast checking	Fast check CRC mode begins the check from container 64 and does not sleep between checking containers. When the fast CRC ends, CRC behavior reverts to the behavior before fast checking was invoked. UNIX: `/usr/openv/pdde/pdcr/bin/crcontrol --crccheckrestart` Windows: *`install_path`*`\Veritas\pdde\Crcontrol.exe --crccheckrestart`
Get the status	UNIX: `/usr/openv/pdde/pdcr/bin/crcontrol --crccheckstate` Windows: *`install_path`*`\Veritas\pdde\Crcontrol.exe --crccheckstate`

To configure data integrity checking behavior by editing the configuration files

1. Use a text editor to open the `contentrouter.cfg` file or the `spa.cfg` file, which control the data integrity checking behavior.

   The files reside in the following directories:

   - UNIX: *`storage_path`*`/etc/puredisk`
   - Windows: *`storage_path`*`\etc\puredisk`

2. To change a parameter, specify a new value.

   See "MSDP data integrity checking configuration parameters" on page 557.

3. Save and close the file.

4. Restart the NetBackup Deduplication Engine and the NetBackup Deduplication Manager.

   You can do this from the **Daemons** tab in the **Activity Monitor**.

MSDP data integrity checking configuration parameters

The configuration file parameters that control the deduplication data integrity checking are in two different configuration files, as follows:

- The `contentrouter.cfg` file.
  The parameters are described in Table 10-3.
  See "About the MSDP contentrouter.cfg file" on page 194.
- The `spa.cfg` file.

The parameters are described in Table 10-3.

Those files reside in the following directories:

- UNIX: *storage_path*/etc/puredisk
- Windows: *storage_path*\etc\puredisk

Warning: Cohesity recommends that you do not disable the data integrity checking. If you do so, NetBackup cannot find and repair or report data corruption.

See "About MSDP data integrity checking" on page 554.

Table 10-3 The contentrouter.cfg file parameters for data integrity checking

Setting	Default	Description
EnableCRCCheck	true	Enable or disable cyclic redundancy checking (CRC) of the data container files. The possible values are true or false. CRC occurs only when no backup, restore, or queue processing jobs are running.
CRCCheckSleepSeconds	5	The time in seconds to sleep between checking containers. The longer the sleep interval, the more time it takes to check containers.
CRCCheckBatchNum	40	The number of containers to check each time. The greater the number of containers, the less time it takes to check all containers, but the more system resources it takes.
ShutdownCRWhenError	false	Stops the NetBackup Deduplication Manager when a data loss is discovered. This parameter is reserved for debugging purposes by Cohesity Support Representatives. The possible values are true or false.
GarbageCheckRemainDCCount	100	The number of containers from failed jobs not to check for garbage. A failed backup or replication job still produces data containers. Because failed jobs are retried, retaining those containers means NetBackup does not have to send the fingerprint information again. As a result, retried jobs consume less time and fewer system resources than when first run.

Table 10-3 The contentrouter.cfg file parameters for data integrity checking *(continued)*

Setting	Default	Description
`CRCDaysToRaiseCritical DataScannedNotification`	7	A critical event is generated if the data is not scanned for CRC on local LSU within a specified number of days. This parameter must be added as a new entry under `DataCheck` for initial use.
`EnableCRCDataScan Notification`	true	Disable the CRC notifications for local LSU. This parameter must be added as a new entry under `DataCheck` for initial use. For example, `EnableCRCDataScanNotification=false`

Table 10-4 spa.cfg file parameters for data integrity checking

Setting	Default	Description
`EnableDataCheck`	`true`	Enable or disable data consistency checking. The possible values are `True` or `False`.
`DataCheckDays`	`14`	The number of days to check the data for consistency. The greater the number of days, the fewer the objects that are checked each day. The greater the number of days equals fewer storage server resources consumed each day.
`EnableDataCheckAlert`	`true`	Enable or disable alerts. If `true`, NetBackup writes a message to the Disk Logs report when it detects a lost data segment. See "NetBackup MSDP log files" on page 767.

About the CRC notifications for local LSU

NetBackup sends the notifications related to the status of CRC for local LSU. You can see these notifications on the NetBackup web UI. You can also use the NetBackup APIs to fetch these notifications. For more information about the APIs, see the *NetBackup API documentation*.

The CRC notifications for local LSU are enabled by default. You can disable it by using the `EnableCRCDataScanNotification` parameter.

See "MSDP data integrity checking configuration parameters" on page 557.

Table 10-5 CRC notifications

Notification	Type	Frequency	Configuration
Some data scanned in the last 24 hrs	Information	Once every 24 hours.	Not configurable
No data scanned in the last 24 hrs	Warning	Once every 24 hours.	Not configurable
No data scanned in the last N days	Critical	Once every 24 hours.	Configurable See "MSDP data integrity checking configuration parameters" on page 557.
Data corruption found	Critical	When the data corruption is found. The notification is not repeated more than once in 24 hours.	Not configurable

About managing MSDP storage read performance

NetBackup provides some control over the processes that are used for read operations. The read operation controls can improve performance for the jobs that read from the storage. Such jobs include restore jobs, duplication jobs, and replication jobs.

In most cases, you should change configuration file options only when directed to do so by a Cohesity support representative.

Defragment the storage

NetBackup includes a process, called *rebasing*, which defragments the backup images in a deduplication pool. Read performance improves when the file segments from a client backup are close to each other on deduplication storage.

See "About MSDP storage rebasing" on page 561.

Decrypt the data on the client rather than the server

The `RESTORE_DECRYPT_LOCAL` parameter in the `pd.conf` file specifies on which host to decrypt and decompress the data during restore operations.

See "About the MSDP pd.conf configuration file" on page 178.

See "MSDP pd.conf file parameters" on page 179.

About MSDP storage rebasing

During an initial backup, NetBackup writes the data segments from a backup to as few container files as possible. Read performance is best when the data segments from a client backup are close to each other on deduplication storage. NetBackup consumes less time finding and reassembling backed up files when their segments are near each other.

However, the data segments in a backup may become scattered across the disk storage each time the client is backed up. Such scattering is a normal consequence of deduplication.

NetBackup includes a process, called *rebasing*, that helps to maintain the data segments in as few container files as possible. Rebasing improves performance for the operations that read from the storage, such as restores and duplications. NetBackup writes all of the data segments from a backup into new container files even though the segments exist on storage already. Future backups then refer to the new copies of those segments rather than the old copies until any changes because of future rebasing. Deduplication rates for the backup jobs that perform rebasing are lower than for the jobs that do not rebase the data.

After the rebasing, NetBackup reclaims the storage space that the rebased data segments used.

Table 10-6 describes the rebasing operations.

Table 10-6 Types of rebasing

Type	Description
Normal backup rebasing	The rebasing that occurs during a backup if the normal rebasing criteria are met, as follows: ■ The container has been rebased within the last 3 months. ■ For that backup, the data segments in the container consume less space than the `FP_CACHE_PERIOD_REBASING_THRESHOLD` value. The `FP_CACHE_PERIOD_REBASING_THRESHOLD` parameter is in the `pd.conf` file. See "MSDP pd.conf file parameters" on page 179. Backup rebasing occurs only for the full backups that pass through the normal MSDP backup process. For example, the NetBackup Accelerator backups do not pass through the MSDP backup process. NetBackup reports backup job completion after the rebasing is completed.

Table 10-6 Types of rebasing *(continued)*

Type	Description
Periodic backup rebasing	The rebasing that occurs during a backup if the periodic rebasing criteria are met, as follows: ■ The container has not been rebased within the last 3 months. ■ For that backup, the data segments in the container consume less space than the `FP_CACHE_REBASING_THRESHOLD` value. The `FP_CACHE_REBASING_THRESHOLD` parameter is in the `pd.conf` file. See "MSDP pd.conf file parameters" on page 179. Backup rebasing occurs only for the full backups that pass through the normal MSDP backup process. For example, the NetBackup Accelerator backups do not pass through the MSDP backup process. NetBackup reports backup job completion after the rebasing is completed.
Server-side rebasing	The storage rebasing that occurs on the server if the rebasing criteria are met. Server-side rebasing includes the deduplicated data that does not pass through the normal MSDP backup process. For example, the NetBackup Accelerator backups do not pass through the MSDP backup process. Some parameters in the `contentrouter.cfg` file control the server-side rebasing behavior. See "MSDP server-side rebasing parameters" on page 562.

MSDP server-side rebasing parameters

Table 10-7 describes the parameters that control server-side rebasing.

See "About MSDP storage rebasing" on page 561.

Usually, you do not need to change parameter values. However, in some cases, you may be directed to change settings by a Cohesity support representative.

The parameters are stored in the `contentrouter.cfg` file.

See "About the MSDP contentrouter.cfg file" on page 194.

Table 10-7 The server-side rebasing parameters

Parameter	Description
`RebaseMaxPercentage`	The maximum percentage of the data segments to be rebased in a file. For any file, if the percentage of the data segments reaches this threshold, the remainder of the data segments are not rebased. By default, this parameter is `RebaseMaxPercentage=5`.
`RebaseMaxTime`	The maximum time span in seconds of data segments to be rebased in a file. If this threshold is reached, NetBackup does not rebase the remainder of the data segments. By default, this parameter is `RebaseMaxTime=150`.
`RebaseMinContainers`	The minimum number of containers in which a file's data segments are stored for the file to be eligible for rebasing. If the number of containers in which a file's data segments are stored is less than `RebaseMinContainers`, NetBackup does not rebase the data segments. By default, this parameter is `RebaseMinContainers=4`.
`RebaseScatterThreshold`	The data locality threshold for a container. If the total size of a file's data segments in a container is less than `RebaseScatterThreshold`, NetBackup rebases all of the file's data segments. By default, this parameter is `RebaseScatterThreshold=64MB`.

About the MSDP data removal process

The data removal process removes the data segments that comprise a NetBackup backup image. Only those segments that are not referred to by a backup image are removed.

The following list describes the data removal process for expired backup images:

- NetBackup removes the image record from the NetBackup catalog.
  NetBackup directs the NetBackup Deduplication Manager to remove the image.
- The deduplication manager immediately removes the image entry in the deduplication catalog and adds a removal request to the NetBackup Deduplication Engine's transaction queue.
  From this point on, the expired backup image is no longer accessible.
- When the NetBackup Deduplication Engine processes the queue, all of the removal requests are processed. A removal request for the image is not generated again.

During the queue processing, the Deduplication Engine reclaims some of the storage space on which the data segments reside. Some is reclaimed during data compaction. If a different backup image requires a data segment, the segment is not removed.

Various internal parameters control whether a container file is compacted.

See "About MSDP container files" on page 523.

If you manually delete an image that has expired within the previous 24 hours, the data becomes garbage. It remains on disk until removed by the next garbage collection process. Garbage collection occurs during data integrity checking.

See "About MSDP data integrity checking" on page 554.

See "Deleting backup images" on page 552.

Resizing the MSDP storage partition

If the volume that contains the deduplication storage is resized dynamically, restart the NetBackup services on the storage server. You must restart the services so that NetBackup can use the resized partition correctly. If you do not restart the services, NetBackup reports the capacity as full prematurely.

To resize the MSDP storage

1. Stop all NetBackup jobs on the storage on which you want to change the disk partition sizes and wait for the jobs to end.
2. Deactivate the media server that hosts the storage server.

   See the *NetBackup Administrator's Guide, Volume I*:

   https://support.cohesity.com/s/article/article-100040135
3. Stop the NetBackup services on the storage server.

   Be sure to wait for all services to stop.
4. Use the operating system or disk manager tools to dynamically increase or decrease the deduplication storage area.
5. Restart the NetBackup services.
6. Activate the media server that hosts the storage server.

   See the *NetBackup Administrator's Guide, Volume I*:

   https://support.cohesity.com/s/article/article-100040135
7. Restart the deduplication jobs.

How MSDP restores work

The following two methods exist to for MSDP restore operations:

Table 10-8 MSDP restore types

Type	Description
Normal restore	The MSDP storage server first *rehydrates* (that is, reassembles) the data. NetBackup then chooses the least busy media server to move the data to the client. (NetBackup chooses the least busy media server from those that have credentials for the NetBackup Deduplication Engine.) The media server `bptm` process moves the data to the client. The following media servers have credentials for the NetBackup Deduplication Engine: ■ The media server that hosts the storage server. Although the media server and the storage server share a host, the storage server sends the data through the media server `bptm` process on that host. ■ A load balancing server in the same deduplication node. See "About MSDP load balancing servers" on page 36. ■ A deduplication server in a different deduplication node that is the target of optimized duplication. See "About the media servers for MSDP optimized duplication within the same domain" on page 124. You can specify the server to use for restores. See "Specifying the restore server" on page 567.
Restore directly to the client	The storage server can bypass the media server and move the data directly to the client. You must configure NetBackup to bypass a media server and receive the restore data directly from the storage server. See " Configuring MSDP restores directly to a client" on page 565. By default, NetBackup decompresses the data on the NetBackup media server except for client direct restore. In that case, the data decompression is done on the client. You can configure NetBackup so that the data is decompressed on the client rather than the storage server. See the `RESTORE_DECRYPT_LOCAL` parameter in the MSDP `pd.conf` file. See "MSDP pd.conf file parameters" on page 179. See "Editing the MSDP pd.conf file" on page 179.

Configuring MSDP restores directly to a client

The NetBackup MSDP storage server can move restore data directly to an MSDP client, bypassing the media server components.

See "How MSDP restores work" on page 565.

To enable restores directly to a client

1. Set the `OLD_VNETD_CALLBACK` option to `YES` on the client. The `OLD_VNETD_CALLBACK` option is stored in the `bp.conf` file on UNIX systems and the registry on Windows systems.

   See "Setting NetBackup configuration options by using the command line" on page 135.

2. On the primary server, run the following command to configure NetBackup to use client-direct restores for the client:

   UNIX: `/usr/openv/netbackup/bin/admincmd/bpclient -client` *`client_name`* `-update -client_direct_restore 2`

   Windows: *`install_path`*`\NetBackup\bin\admincmd\bpclient -client` *`client_name`* `-update -client_direct_restore 2`

About restoring files at a remote site

If you use optimized duplication to copy images from a local site to a remote site, you can restore from the copies at the remote site to clients at the remote site. To do so, use a server-directed restore or a client-redirected restore, which restores files to a client other than the original client.

Information about how to redirect restores is in a different guide.

See "Managing client restores" in the *NetBackup Administrator's Guide, Volume I*.

You may have to configure which media server performs the restore. In optimized duplication, the media server that initiates the duplication operation becomes the write host for the new image copies. The write host restores from those image copies. If the write host is at the local site, it restores from those images at the remote site to the alternate client at the remote site. That host reads the image across the WAN and then writes the image back across the WAN to the alternate client. In this case, you can specify that the media server at the remote site as the restore server.

About restoring from a backup at a target primary domain

While it is possible to restore a client directly by using the images in the target primary domain, do so only in a disaster recovery situation. In this discussion, a disaster recovery situation is one in which the originating domain no longer exists and clients must be recovered from the target domain

Table 10-9 Client restores in disaster recovery scenarios

Disaster recovery scenario	Does client exist?	Description
Scenario 1	Yes	Configure the client in another domain and restore directly to the client.
Scenario 2	No	Create the client in the recovery domain and restore directly to the client. This is the most likely scenario.
Scenario 3	No	Perform an alternate client restore in the recovery domain.

The steps to recover the client are the same as any other client recovery. The actual steps depend on the client type, the storage type, and whether the recovery is an alternate client restore.

For restores that use Granular Recovery Technology (GRT), an application instance must exist in the recovery domain. The application instance is required so that NetBackup has something to recover to.

Specifying the restore server

NetBackup may not use the backup server as the restore server for deduplicated data.

See "How MSDP restores work" on page 565.

You can specify the server to use for restores. The following are the methods that specify the restore server:

- Always use the backup server. Two methods exist, as follows:
  - Use NetBackup **Host properties** to specify a **Media host override** server. All restore jobs for any storage unit on the original backup server use the media server you specify. Specify the same server for the **Restore server** as for the **Original backup server**.
    See "Forcing restores to use a specific server" in the NetBackup Administrator's Guide, Volume I.
    This procedure sets the `FORCE_RESTORE_MEDIA_SERVER` option. Configuration options are stored in the `bp.conf` file on UNIX systems and the registry on Windows systems.
  - Create the touch file `USE_BACKUP_MEDIA_SERVER_FOR_RESTORE` on the NetBackup primary server in the following directory:
    UNIX: `usr/openv/netbackup/db/config`
    Windows: *`install_path`*`\Veritas\Netbackup\db\config`

This global setting always forces restores to the server that did the backup. It applies to all NetBackup restore jobs, not just deduplication restore jobs. If this touch file exists, NetBackup ignores the `FORCE_RESTORE_MEDIA_SERVER` and `FAILOVER_RESTORE_MEDIA_SERVER` settings.

- Always use a different server.
  Use NetBackup **Host Properties** to specify a **Media host override** server.
  See the previous explanation about **Media host override**, except: Specify the different server for the **Restore server**.
- A single restore instance. Use the `bprestore` command with the `-disk_media_server` option.
  Restore jobs for each instance of the command use the media server you specify.
  See the NetBackup Commands Reference Guide.

Enabling extra OS STIG hardening on WORM storage server instance

The Security Technical Implementation Guides (STIGs) provide technical guidance for increasing the security of information systems and software to help prevent malicious computer attacks. This type of security is also referred to as hardening.

OS STIG hardening rules are automatically enabled on primary, media, and storage server instances. These rules are based on the following profile from the Defense Information Systems Agency (DISA):

STIG for Red Hat Enterprise Linux Server

You can enable extra OS STIG hardening for increased security. The additional rules add protection to the `sshd` process and enforce stricter password policies.

Note the following about enabling extra OS STIG hardening:

- This command does not allow individual rule control.
- Once the option is enabled, it cannot be disabled.
- Before the extra rules are enabled on the instance, you can have unlimited concurrent SSH sessions. After OS STIG hardening is enabled, the maximum number of concurrent SSH sessions is limited to 10.

To enable extra OS STIG hardening

1. Open an SSH session to the instance as the **msdpadm** user (Flex) or **appadmin** user (Flex Scale).
2. Run the following command:

```
setting STIG enable-ondemand-hardening
```

Using multiple MSDP nodes for multistream backups on MSDP cluster

In the MSDP cluster environments, balancing the load across multiple nodes is crucial for maintaining optimal performance. MSDP has either a single node configuration or a cluster configuration. MSDP cluster configurations such as Flex Scale, Cloud Scale, and MSDP volume groups may have multiple nodes. These clustered configurations allow for enhanced scalability and performance for multistream backups.

Starting with NetBackup 11.0, a new policy attribute **Use multiple MSDP nodes** is introduced to improve load balancing for multistream backups on MSDP cluster storage servers. This feature enables the distribution of backup streams across the multiple MSDP nodes, optimizing throughput, and reducing potential congestion on individual nodes.

Before NetBackup 11.0, the policy name prefix MSDPLB+ was used to enable the use of multiple MSDP nodes for multistream backups. While functional, the MSDPLB+ naming convention was not always the most convenient, and prompted the introduction of the new policy attribute in NetBackup 11.0.

Consider the following points when using this option:

- This policy attribute is designed to use only for multiple stream backups on the MSDP cluster storage servers. For a single-node MSDP storage server, this attribute has no effect.
- This policy attribute is not available for the following policy types. Some of these policy types already have more advanced features that handle the distribution of backup streams across multiple MSDP nodes. For others, using multiple MSDP nodes is not applicable due to the nature of the policies.
  - Oracle
  - Epic-large-file
  - Universal-share
  - MSDP-object-store
- The media server must be running NetBackup 11.0 or later to use this option. If the storage server is configured with the media servers of an earlier version, the backup job fails with the job status code 213. However, for environments running earlier versions, the MSDPLB+ policy prefix can still be used to enable load balancing across multiple MSDP nodes.

- For smaller backups, we recommend not enabling this option. Without this option enabled, deduplication on the same node is the most efficient way to minimize storage and network utilization.
- If the client data is large and the backup job cannot finish within the backup window, enable this option to use multiple MSDP nodes for parallel data processing, reducing backup time.
- Some client applications might not send backup data to NetBackup in a same order as the previous backups. In such cases, MSDP may require more storage and network resources across nodes to manage the data streams. Enabling this option for these types of applications may reduce system efficiency.

See "Enabling the media server and MSDP engine affinity in the MSDP cluster" on page 570.

Enabling the media server and MSDP engine affinity in the MSDP cluster

In an MSDP cluster environment, both the MSDP engine and the media server can run on the same node. By default, NetBackup does not maintain the relationship between the two. When a backup job runs, the media server transfers the data to one of the MSDP engines. If the media server and MSDP engine are not on the same node, the data is transferred over the network, which may require more network bandwidth.

If you have a concern about the network bandwidth, you can enable the media server affinity setting. By enabling the media server affinity, NetBackup tries to select the media server and MSDP engine on the same node for backup jobs, reducing network traffic.

To enable the media server and MSDP engine affinity

1. Run the following command on the NetBackup primary server to enable the setting:

```
/usr/openv/netbackup/bin/admincmd/nbemmcmd -changesetting
-machinename <primary_server> -ALLOW_MEDIA_SERVER_AFFINITY 1
```

2. Run the following command to check the current settings:

```
/usr/openv/netbackup/bin/admincmd/nbemmcmd -listsettings
-machinename <primary_server>
```

Chapter 11

Recovering MSDP

This chapter includes the following topics:

- Recovering the NetBackup MSDP catalog
- MSDP catalog recovery prerequisites
- Synchronizing the date and time of the NetBackup primary server and the MSDP storage server
- Determining the catalog backup for recovery
- Recovering the MSDP catalog automatically
- Recovering the MSDP catalog manually
- Recovering from an MSDP storage server disk failure
- Recovering from an MSDP storage server failure
- Recovering the MSDP storage server after NetBackup catalog recovery

Recovering the NetBackup MSDP catalog

The NetBackup Media Server Deduplication Pool (MSDP) catalog contains critical metadata for deduplication operations. If the catalog becomes corrupted or lost, recovery is essential to restore deduplication functionality. NetBackup provides the following two primary recovery options for the MSDP catalog:

- Automatic recovery from shadow copy: NetBackup restores the catalog from the latest shadow copy and replays the transaction log to ensure it is current. Manual recovery from a shadow copy is also available if needed.
- Recovery from backup: If shadow copies are unavailable and a valid MSDP catalog backup exists, you can recover from the backup. This option should

only be used as a last resort and ideally under Cohesity Support guidance to avoid data loss.

To configure and recover the MSDP catalog For Flex WORM, CloudScale, and Flex Scale, See "Restoring from the external MSDP catalog backup" on page 723.

Caution: MSDP catalog disaster recovery (DR) is designed to recover the MSDP catalog only when both the MSDP catalog and its shadow catalog are missing or completely corrupted. This recovery process is highly specific and should not be used for any other scenario. Using either the automated or manual MSDP catalog recovery procedure outside of these circumstances may result in data loss or inaccessible backups.

You can perform the manual recovery or automated recovery of the MSDP catalog:

- Automated MSDP Catalog Recovery
  This method uses NetBackup utilities to streamline the recovery process, typically restoring from the most recent full and incremental backups. Automated recovery is suitable for straightforward scenarios and minimizes manual intervention.
- Manual MSDP Catalog Recovery
- Manual recovery provides granular control over the restoration process. It is required when you need to recover from a specific backup (not the latest), import images from the storage server, or address complex recovery scenarios.
  This method involves verifying backup files, initializing MSDP, restoring to shadow directories, and synchronizing catalogs.

MSDP catalog recovery prerequisites

Before you begin an MSDP catalog recovery, the following prerequisites must be satisfied:

- The storage server host must be functional.
- The MSDP storage server must be functional.
- The catalog directories must exist. The directories can be empty, but the file systems on which they reside must be healthy, their mount points must exist, and they must be mounted. The following are the default locations for the catalog directories: If the paths differ on your system, ensure that the mount points exist.
  - `/database_path/databases`
  - `/storage_path/var`
  - `/storage_path/catalogshadow`

If the setup was upgraded from an older version:
`/database_path/databases/catalogshadow`

- The values of the following three parameters in the spa.cfg file section:variable pairs) must be the same in the recovery environment as in the backup from which you recover:
  - `StorageDatabase:CatalogShadowPath`
  - `StorageDatabase:Path`
  - `Paths:Var`
- The NetBackup primary server must be functional.
- The NetBackup catalog must be available and uncorrupted.
- The storage unit that contains the MSDP catalog backup must be available.
- Other users of the host must not stop or start the NetBackup deduplication services for the duration of the recovery.

Some of the recovery is resource intensive; Cohesity recommends that you quiesce everything else on the host during the recovery.

If the NetBackup deduplication services are stopped or started manually during this procedure, data loss can occur.

Synchronizing the date and time of the NetBackup primary server and the MSDP storage server

Ensure that the operating system date and time of the storage server is consistent with the time of the NetBackup primary server. If the date and time are not reasonably synchronized, some of the NetBackup commands may fail with a "no entity found" message or more recent catalog backups might not be available. Also, some deduplication images could be incorrectly removed.

Determining the catalog backup for recovery

In most cases, the backup images to use for the recovery are the most recent Full and all subsequent Incremental deduplication catalog backups.

If you need to select a different set of backups, use the manual recovery method, which allows you to specify a date range that contains the desired backup images. Refer to the *NetBackup Bare Metal Restore Administrator's Guide*, *NetBackup Web UI Administrator's Guide*, or the `bprestore` command in the *NetBackup Commands Reference Guide*.

You must choose an MSDP catalog backup image that was created from the current deduplication storage server. Catalog images from a different deduplication storage server cannot be used for recovery.

Recovering the MSDP catalog automatically

You cannot use the automated recovery method in the following scenarios:

- If you want to import images from this storage server, for example, if you have recovered the NetBackup catalog (the primary server's catalog) and have not imported the images from this storage server, you must use the manual steps or you will not have any images to import (from this deduplication storage server) after the deduplication catalog recovery process has finished.
- If you want to recover from a backup other than the most recent. You must use the manual method so that you can specify the date range that contains the desired backup images.

Table 11-1 Automated MSDP catalog recovery

Step	Task	Description
1	Ensure that the pre-requisites are met.	See "MSDP catalog recovery prerequisites" on page 572.
2	Log on to the MSDP storage server that you want to recover	The MSDP catalog recovery steps require NetBackup commands, MSDP commands, and operating system commands. Open a command or shell window on the NetBackup MSDP storage server to run the required NetBackup, MSDP, and operating system commands for recovery.
3	Synchronize the date and time of the NetBackup primary server and the MSDP storage server.	See "Synchronizing the date and time of the NetBackup primary server and the MSDP storage server " on page 573.
4	Determine the catalog backup from which to recover.	See "Determining the catalog backup for recovery " on page 573.
5	Recover the MSDP catalog from a backup.	See "Recovering the MSDP Catalog from a backup" on page 575.

Recovering the MSDP Catalog from a backup

If you need to import images from the deduplication storage server, do not use the automated recovery process. For example, if you have already restored the NetBackup primary server catalog but have not imported images from this storage server, you must follow the manual recovery steps. Using the automated process in this scenario will result in no images available for import after catalog recovery.

The NetBackup **drcontrol** utility offers an automated option to recover the MSDP catalog from the most recent FULL backup and all subsequent INCR backups. For recovery from a specific date range, refer to the documentation for the bprestore command.

To recover the MSDP Catalog from a backup

1. Run the `drcontrol` utility on the MSDP storage server:
   - UNIX: `/usr/openv/pdde/pdcr/bin/drcontrol --auto_recover_DR --policy policy_name [-client client_name]`
   - Windows: `install_path\Veritas\pdde\drcontrol --auto_recover_DR --policy policy_name [-client client_name]`
     Specify the `client_name` if the value returned by `bpgetconfig CLIENT_NAME` differs from the backup image client name. This is typically required if hostnames or domain names were not used consistently.
2. Monitor the command output for progress messages and status codes.
3. After completion, the utility displays the path to the log file for review.

Recovering the MSDP catalog manually

Perform the following steps to recover the MSDP catalog manually:

Table 11-2 Automated MSDP catalog recovery

Step	Task	Description
1	Ensure that the pre-requisites are met.	See "MSDP catalog recovery prerequisites" on page 572.
2	Log on to the MSDP storage server that you want to recover	The MSDP catalog recovery steps require NetBackup commands, MSDP commands, and operating system commands. Open a command or shell window on the NetBackup MSDP storage server to run the required NetBackup, MSDP, and operating system commands for recovery.

Table 11-2 Automated MSDP catalog recovery *(continued)*

Step	Task	Description
3	Synchronize the date and time of the NetBackup primary server and the MSDP storage server.	See "Synchronizing the date and time of the NetBackup primary server and the MSDP storage server " on page 573.
4	Determine the catalog backup from which to recover.	See "Determining the catalog backup for recovery " on page 573.
5	Verify the catalog backup files.	See "Verifying the catalog backup files" on page 576.
6	Initialize MSDP for the catalog recovery.	See "Initializing MSDP for catalog recovery " on page 578.
7	Restore an MSDP catalog backup to the catalog shadow directory.	See "Restoring an MSDP catalog backup to the catalog shadow directory" on page 578.
8	Recover the MSDP catalog from a shadow copy.	See "Recovering the MSDP Catalog from a shadow copy" on page 579.
9	Synchronize the MSDP catalog with the NetBackup catalog.	See "Synchronize the MSDP catalog with the NetBackup catalog" on page 579.
10	Set the NetBackup Deduplication Engine to refuse jobs.	See "Setting the NetBackup Deduplication Engine to refuse jobs" on page 580.
11	Check MSDP data integrity manually.	See "Checking MSDP data integrity manually " on page 581.
12	Process the MSDP transaction queue manually.	See "Process the MSDP transaction queue manually" on page 582.
13	Set the NetBackup Deduplication Engine to accept jobs.	See "Setting the NetBackup Deduplication Engine to accept jobs" on page 582.

Verifying the catalog backup files

Before manually recovering the MSDP catalog from a backup, it is essential to verify that the required backup files exist. You can verify either the most recent backup or a backup from a specific date.

To Verify the Most Recent Catalog Backup Files

1. Check the client name.

   If the client name differs from the value returned by the b`pgetconfig CLIENT_NAME` command, use the `-client host_name` option.

2. Run the following command on the MSDP storage server:

   - UNIX:`/usr/openv/pdde/pdcr/bin/drcontrol --list_files --policy policy_name [-client client_name]`
   - Windows:`install_path\Veritas\pdde\drcontrol --list_files --policy policy_name [-client client_name]`

3. Examine the log file specified in the command output. If the log shows the catalog directories and files, you can proceed to initializing MSDP for catalog recovery.

To Verify the Catalog Backup Files from a Specific Date

1. Run the following command on the MSDP storage server to list files backed up on a specific date:

   - UNIX:`/usr/openv/netbackup/bin/bplist -C client_name -k policy_name -R -l -b -t 0 -s mm/dd/yyyy 00:00:00 -e mm/dd/yyyy 23:59:59`
   - Windows:`install_path\Veritas\NetBackup\bin\bplist -C client_name -k policy_name -R -l -b -t 13 -s mm/dd/yyyy 00:00:00 -e mm/dd/yyyy 23:59:59`

   Use the same date for both the start (-s) and end (-e) options. The date and time format depends on your locale.

2. Examine the output.

   If the output shows the catalog directories and files for the correct date, proceed to the next step.

3. Verify the backup image. Run the following command to confirm the backup is valid:

   - UNIX: `/usr/openv/netbackup/bin/admincmd/bpverify -local -policy policy_name -client client_name -s mm/dd/yyyy 00:00:00 -e mm/dd/yyyy 23:59:59`
   - Windows: `install_path\Veritas\NetBackup\bin\admincmd\bpverify -local -policy policy_name -client client_name -s mm/dd/yyyy 00:00:00 -e mm/dd/yyyy 23:59:59`

Use the same date for both the start and end options.

4 Review the verification output.

If the command output indicates a successful verification and displays the catalog directories and files, you can proceed to initializing MSDP for catalog recovery.

Initializing MSDP for catalog recovery

As part of the manual MSDP catalog recovery process, you must initialize MSDP for recovery. This step stops the necessary services and moves the existing catalog files to prepare for restoration.

To initialize MSDP for catalog recovery

- Run the following command on the MSDP storage server:
  - UNIX: `/usr/openv/pdde/pdcr/bin/drcontrol --initialize_DR --policy policy_name`
  - Windows: `install_path\Veritas\pdde\drcontrol --initialize_DR --policy policy_name`

Restoring an MSDP catalog backup to the catalog shadow directory

To prepare for catalog recovery, restore the backup images to the `database_path/databases/catalogshadow` directory.

Note: By default, storage_path and database_path are the same, but a different database_path may be specified during installation.

Always restore from an MSDP catalog backup image created on the current MSDP storage server. Attempting to recover from a backup made on a different MSDP storage server can result in data loss.

To restore an MSDP catalog backup to the catalog shadow directory

- Restore the deduplication catalog image by using one of the following methods:
  - The `drcontrol` command. This option restores from the most recent backup.
    UNIX:`/usr/openv/pdde/pdcr/bin/drcontrol --recover_last_catalog_image --policy policy_name`
    Windows:`install_path\Veritas\pdde\drcontrol -recover_last_catalog_image --policy policy_name`
  - The `drcontrol` command. This option restores from the most recent backup.

UNIX:`/usr/openv/netbackup/bin/bprestore -C client_name -p policy_name [-s mm/dd/yyyy 00:00:00 -e mm/dd/yyyy 23:59:59]`
Windows:`install_path\Veritas\NetBackup\bprestore --recover_last_catalog_image --policy policy_name [-s mm/dd/yyyy 00:00:00 -e mm/dd/yyyy 23:59:59]`
Specify both the start (-s) and end (-e) options using the same date. The date and time format depends on your locale.

- Backup, Archive, and Restore interface:
  Use this option to restore from a backup other than the most recent.

Recovering the MSDP Catalog from a shadow copy

After you restore the MSDP catalog from a backup to the catalog shadow directories, you must recover the catalog from the catalog shadow directory.

To recover the MSDP Catalog from a shadow copy

1. Start the NetBackup Deduplication Manager (`spad`):

   UNIX:`/usr/openv/pdde/pdconfigure/pdde start spad`

   Windows:`install_path\Veritas\pdde\spad --start`

2. Recover the catalog from the shadow copy:

   UNIX:`/usr/openv/pdde/pdcr/bin/cacontrol --catalog recover all`

   Windows:`<install_path>\Veritas\pdde\cacontrol --catalog recover all`

3. Stop the NetBackup Deduplication Manager (`spad`):

   UNIX: `/usr/openv/pdde/pdconfigure/pdde stop spad`

   Windows: `install_path\Veritas\pdde\spad --stop`

Synchronize the MSDP catalog with the NetBackup catalog

When you recover the MSDP catalog from a backup, you must synchronize it with the NetBackup catalog to ensure consistency.

To synchronize the MSDP catalog with the NetBackup catalog

1. Run the following command to synchronize the catalog.

   The command deletes the metadata in the MSDP catalog that does not have a corresponding backup image in the NetBackup catalog.

   UNIX:`/usr/openv/pdde/pdcr/bin/dedupecatutil --FORCE_DEDUP_METADATA_CONSISTENCY [--num_threads N]`

   Windows: <install_path>\Veritas\pdde\dedupecatutil --FORCE_DEDUP_METADATA_CONSISTENCY [--num_threads N]

   The `--num_threads N` option lets you specify the number of processor threads for the task. By default `--num_threads` is set to `1`.

   After this command completes successfully, the existing shadow catalog becomes invalid.

2. Run the following command to create a new catalog shadow copy:

   UNIX: `/usr/openv/pdde/pdcr/bin/drcontrol --refresh_shadow_catalog`

   Windows:`install_path\Veritas\pdde\drcontrol --refresh_shadow_catalog`

Setting the NetBackup Deduplication Engine to refuse jobs

When recovering the MSDP catalog from a backup, you may need to configure the NetBackup Deduplication Engine to refuse jobs. The Deduplication Engine must remain active, but it cannot process jobs during the recovery process.

To set the NetBackup Deduplication Engine to refuse jobs

1. Start the NetBackup Deduplication Engine (`spoold`), as follows:

   UNIX:`/usr/openv/pdde/pdconfigure/pdde start spoold`

2. Set the NetBackup Deduplication Engine to refuse to accept jobs, as follows:

   - UNIX:

     ```
     /usr/openv/pdde/pdcr/bin/crcontrol -m GET=No
     /usr/openv/pdde/pdcr/bin/crcontrol -m PUT=No
     ```

   - Windows:

     ```
     install_path\Veritas\pdde\crcontrol -m GET=No
     install_path\Veritas\pdde\crcontrol -m PUT=No
     ```

3 Verify that the NetBackup Deduplication Manager (`spad`) and the NetBackup Deduplication Engine (`spoold`) are active. Use the `bpps` command.

UNIX:`/usr/openv/netbackup/bin/bpps`

Windows:`install_path\Veritas\NetBackup\bin\bpps`

4 If `spad` or `spoold` are not active, start them:

- UNIX:
  ```
  /usr/openv/pdde/pdconfigure/pdde spad start
  /usr/openv/pdde/pdconfigure/pdde spoold start
  ```
- Windows:
  ```
  install_path\Veritas\pdde\spad --start
  install_path\Veritas\pdde\spoold --start
  ```

Checking MSDP data integrity manually

When you recover the MSDP catalog from a backup, you must check the integrity of the MSDP data. The data integrity process also resolves any data integrity issues.

To check MSDP data integrity

1 Run the following command to check the integrity of the MSDP data:

UNIX:`/usr/openv/pdde/pdcr/bin/pddecfg -a startdatafullcheck`

Windows: `install_path\Veritas\pdde\pddecfg -a startdatafullcheck`

2 When the following command lists Busy as No, the data integrity check has been completed.

UNIX:`/usr/openv/pdde/pdcr/bin/pddecfg -a getdataintegritycheck`

Windows:`install_path\Veritas\pdde\pddecfg -a getdataintegritycheck`

3 Open the data integrity check log file in a text editor.

UNIX:`/storage_path/log/spad/OnlineCheck.log`

Windows:`storage_path\log\spad\OnlineCheck.log`

4 Examine the log file for a message similar to the following that corresponds to the date and time that you invoked the command in step 1:

```
Finished sending Data Check POs for data selection[2], cursor=
```

If the line exists, continue to the next step.

5 Run the following command to check if any data container corruption issues exist:

UNIX:`/usr/openv/netbackup/bin/admincmd/bperror -disk`

Windows:`install_path/Veritas\NetBackup\bin\admincmd\bperror -disk`

Process the MSDP transaction queue manually

After recovering the MSDP catalog from a backup, you must manually process the MSDP transaction queue.

To process the MSDP transaction queue manually

1 On the MSDP storage server, run the following command:

UNIX:`/usr/openv/pdde/pdcr/bin/crcontrol --processqueue`

Windows:`install_path\Veritas\pdde\Crcontrol.exe --processqueue`

2 To determine if the queue processing is still active, run the following command:

UNIX:`/usr/openv/pdde/pdcr/bin/crcontrol --processqueueinfo`

Windows:`install_path\Veritas\pdde\Crcontrol.exe --processqueueinfo`

If the output shows Busy: yes, the queue is still active.

3 To examine the results, run the following command (number 1 not lowercase letter l):

UNIX: `/usr/openv/pdde/pdcr/bin/crcontrol --dsstat 1`

Windows:`install_path\Veritas\pdde\Crcontrol.exe --dsstat 1`

The command may run for a long time; if you omit the 1, results return more quickly but they are not as accurate.

Setting the NetBackup Deduplication Engine to accept jobs

After recovering the MSDP catalog from a backup, you may need to configure the NetBackup Deduplication Engine to accept jobs.

To set the NetBackup Deduplication Engine to accept jobs

1. Start the NetBackup services on the storage serve.

   UNIX:`/usr/openv/netbackup/bin/bp.start_all`

   Windows:`install_path\Program Files\Veritas\NetBackup\bin\bpup`

2. Set the NetBackup Deduplication Engine (spoold) to accept jobs, as follows:

   UNIX:

   ```
   /usr/openv/pdde/pdcr/bin/crcontrol –m GET=Yes
   ```

   ```
   /usr/openv/pdde/pdcr/bin/crcontrol –m PUT=Yes
   ```

   Windows:

   ```
   install_path\Veritas\pdde\crcontrol –m GET=Yes
   ```

   ```
   install_path>\Veritas\pdde\crcontrol –m PUT=Yes
   ```

Recovering from an MSDP storage server disk failure

If recovery mechanisms do not protect the disk on which the NetBackup software resides, the deduplication storage server configuration is lost if the disk fails. This topic describes how to recover from a system disk or program disk failure where the disk was not backed up.

Note: This procedure describes recovery of the disk on which the NetBackup media server software resides not the disk on which the deduplicated data resides. The disk may or may not be the system boot disk.

After recovery, your NetBackup deduplication environment should function normally. Any valid backup images on the deduplication storage should be available for restores.

Cohesity recommends that you use NetBackup to protect the deduplication storage server system or program disks. You then can use NetBackup to restore that media server if the disk on which NetBackup resides fails and you have to replace it.

Table 11-3 Process to recover from media server disk failure

Step	Task	Procedure
Step 1	Replace the disk.	If the disk is a system boot disk, also install the operating system. See the hardware vendor and operating system documentation.

Table 11-3 Process to recover from media server disk failure *(continued)*

Step	Task	Procedure
Step 2	Mount the storage.	Ensure that the storage and database are mounted at the same locations. See the storage vendor's documentation.
Step 3	Install and license the NetBackup media server software.	See *NetBackup Installation Guide for UNIX and Windows*: https://support.cohesity.com/s/article/article-100040135
Step 4	Delete the deduplication host configuration file	Each load balancing server contains a deduplication host configuration file. If you use load balancing servers, delete the deduplication host configuration file from those servers. See "Deleting an MSDP host configuration file" on page 200.
Step 5	Delete the credentials on deduplication servers	If you have load balancing servers, delete the NetBackup Deduplication Engine credentials on those media servers. See "Deleting credentials from a load balancing server" on page 541.
Step 6	Add the credentials to the storage server	Add the NetBackup Deduplication Engine credentials to the storage server. See "Adding NetBackup Deduplication Engine credentials" on page 540.
Step 7	Get a configuration file template	If you did not save a storage server configuration file before the disk failure, get a template configuration file. See "Saving the MSDP storage server configuration" on page 196.
Step 8	Edit the configuration file	See "Editing an MSDP storage server configuration file" on page 197.
Step 9	Configure the storage server	Configure the storage server by uploading the configuration from the file you edited. See "Setting the MSDP storage server configuration" on page 198.
Step 10	Add load balancing servers	If you use load balancing servers in your environment, add them to your configuration. See "Adding an MSDP load balancing server" on page 174.

Recovering from an MSDP storage server failure

To recover from a permanent failure of the storage server host computer, use the process that is described in this topic.

NetBackup recommends that you consider the following items before you recover:

- The new computer must use the same byte order as the old computer.

Warning: If the new computer does not use the same byte order as the old computer, you cannot access the deduplicated data. In computing, endianness describes the byte order that represents data: big endian and little endian. For example, SPARC processors and Intel processors use different byte orders. Therefore, you cannot replace an Oracle Solaris SPARC host with an Oracle Solaris host that has an Intel processor.

- Cohesity recommends that the new computer use the same operating system as the old computer.
- Cohesity recommends that the new computer use the same version of NetBackup as the old computer.
  If you use a newer version of NetBackup on the new computer, ensure that you perform any data conversions that may be required for the newer release.
  If you want to use an older version of NetBackup on the replacement host, contact your Cohesity support representative.

Table 11-4 Recover from an MSDP storage server failure

Step	Task	Procedure
Step 1	Delete the storage units that use the disk pool	See the NetBackup Administrator's Guide, Volume I.
Step 2	Delete the disk pool	See "Deleting a Media Server Deduplication Pool" on page 550.
Step 3	Delete the deduplication storage server	See "Deleting an MSDP storage server" on page 538.
Step 4	Delete the deduplication host configuration file	Each load balancing server contains a deduplication host configuration file. If you use load balancing servers, delete the deduplication host configuration file from those servers. See "Deleting an MSDP host configuration file" on page 200.
Step 5	Delete the credentials on deduplication servers	If you have load balancing servers, delete the NetBackup Deduplication Engine credentials on those media servers. See "Deleting credentials from a load balancing server" on page 541.

Table 11-4 Recover from an MSDP storage server failure *(continued)*

Step	Task	Procedure
Step 6	Configure the new host so it meets deduplication requirements	When you configure the new host, consider the following: ■ You can use the same host name or a different name. ■ You can use the same **Storage Path** or a different **Storage Path**. If you use a different **Storage Path**, you must move the deduplication storage to that new location. ■ If the **Database Path** on the original host is different that the **Storage Path**, you can do one of the following: ■ You can use the same **Database Path**. ■ You can use a different **Database Path**. If you do, you must move the deduplication database to the new location. ■ You do not have to continue to use a different **Database Path**. You can move the `databases` directory into the **Storage Path** and then specify only the **Storage Path** when you configure the storage server. ■ You can use the host's default network interface or specify a network interface. If the original host used a specific network interface, you do not have to use the same interface name. ■ If you had configured the previous MSDP storage server to use MSDP Encryption using KMS service, you must use the same configuration for the new MSDP storage server. See "About MSDP storage servers" on page 35. See "About MSDP server requirements" on page 37.
Step 7	Connect the storage to the host	Use the storage path that you configured for this replacement host. See the computer or the storage vendor's documentation.
Step 8	Install the NetBackup media server software on the new host	See the NetBackup Installation Guide.
Step 9	Reconfigure deduplication	You must use the same credentials for the NetBackup Deduplication Engine. See "Configuring media server deduplication in NetBackup" on page 63.
Step 10	Import the backup images.	See the NetBackup Administrator's Guide, Volume I. **Note:** Perform importing of the backup images only when NetBackup catalog is not present; otherwise, use the `bpimage` command to update the storage server names and disk pool names for catalog backup images.

Recovering the MSDP storage server after NetBackup catalog recovery

If a disaster requires a recovery of the NetBackup catalog, you must set the storage server configuration after the NetBackup catalog is recovered.

See “Setting the MSDP storage server configuration” on page 198.

Cohesity recommends that you save your storage server configuration.

See “Save the MSDP storage server configuration” on page 55.

Information about recovering the primary server is available.

See the *NetBackup Troubleshooting Guide*.

Chapter 12

Replacing MSDP hosts

This chapter includes the following topics:

- Replacing the MSDP storage server host computer

Replacing the MSDP storage server host computer

If you replace the deduplication storage server host computer, use these instructions to install NetBackup and reconfigure the deduplication storage server. The new host cannot host a deduplication storage server already.

Reasons to replace the computer include a lease swap or perhaps the current deduplication storage server computer does not meet your performance requirements.

NetBackup recommends that you consider the following items before you recover:

- The new computer must use the same byte order as the old computer.

  Warning: If the new computer does not use the same byte order as the old computer, you cannot access the deduplicated data. In computing, endianness describes the byte order that represents data: Big endian and little endian. For example, SPARC processors and Intel processors use different byte orders. Therefore, you cannot replace an Oracle Solaris SPARC host with an Oracle Solaris host that has an Intel processor.

- Cohesity recommends that the new computer use the same operating system as the old computer.
- Cohesity recommends that the new computer use the same version of NetBackup as the old computer.
  If you use a newer version of NetBackup on the new computer, ensure that you perform any data conversions that may be required for the newer release.

If you want to use an older version of NetBackup on the replacement host, contact your Cohesity support representative.

Table 12-1 Replacing an MSDP storage server host computer

Step	Task	Procedure
Step 1	Expire the backup images	Expire all backup images that reside on the deduplication disk storage. **Warning:** Do not delete the images. They are imported back into NetBackup later in this process. If you use the `bpexpdate` command to expire the backup images, use the `-nodelete` parameter. See the NetBackup Administrator's Guide, Volume I.
Step 2	Delete the storage units that use the disk pool	See the NetBackup Administrator's Guide, Volume I.
Step 3	Delete the disk pool	See "Deleting a Media Server Deduplication Pool" on page 550.
Step 4	Delete the deduplication storage server	See "Deleting an MSDP storage server" on page 538.
Step 5	Delete the deduplication host configuration file	Each load balancing server contains a deduplication host configuration file. If you use load balancing servers, delete the deduplication host configuration file from those servers. See "Deleting an MSDP host configuration file" on page 200.
Step 6	Delete the credentials on deduplication servers	If you have load balancing servers, delete the NetBackup Deduplication Engine credentials on those media servers. See "Deleting credentials from a load balancing server" on page 541.

Table 12-1 Replacing an MSDP storage server host computer *(continued)*

Step	Task	Procedure
Step 7	Configure the new host so it meets deduplication requirements	When you configure the new host, consider the following: ■ You can use the same host name or a different name. ■ You can use the same **Storage Path** or a different **Storage Path**. If you use a different **Storage Path**, you must move the deduplication storage to that new location. ■ If the **Database Path** on the original host is different that the **Storage Path**, you can do one of the following: ■ You can use the same **Database Path**. ■ You can use a different **Database Path**. If you do, you must move the deduplication database to the new location. ■ You do not have to continue to use a different **Database Path**. You can move the `databases` directory into the **Storage Path** and then specify only the **Storage Path** when you configure the storage server. ■ You can use the host's default network interface or specify a network interface. If the original host used a specific network interface, you do not have to use the same interface name. ■ If you had configured the previous MSDP storage server to use MSDP Encryption using KMS service, you must use the same configuration for the new MSDP storage server. See "About MSDP storage servers" on page 35. See "About MSDP server requirements" on page 37.
Step 8	Connect the storage to the host	Use the storage path that you configured for this replacement host. See the computer or the storage vendor's documentation.
Step 9	Install the NetBackup media server software on the new host	See the NetBackup Installation Guide.
Step 10	Reconfigure deduplication	See "Configuring media server deduplication in NetBackup" on page 63.
Step 11	Import the backup images	See the NetBackup Administrator's Guide, Volume I.

Chapter 13

Uninstalling MSDP

This chapter includes the following topics:

- About uninstalling MSDP
- Deactivating MSDP

About uninstalling MSDP

You cannot uninstall media server deduplication components separately from NetBackup. The deduplication components are installed when you install NetBackup software, and they are uninstalled when you uninstall NetBackup software.

Other topics describe related procedures, as follow:

- Reconfigure an existing deduplication environment.
  See "Changing the MSDP storage server name or storage path" on page 536.
- Deactivate deduplication and remove the configuration files and the storage files.
  See "Deactivating MSDP" on page 591.

Deactivating MSDP

You cannot remove the deduplication components from a NetBackup media server. You can disable the components and remove the deduplication storage files and the catalog files. The host remains a NetBackup media server.

This process assumes that all backup images that reside on the deduplication disk storage have expired.

Warning: If you remove deduplication and valid NetBackup images reside on the deduplication storage, data loss may occur.

Table 13-1 Remove MSDP

Step	Task	Procedure
Step 1	Remove client deduplication	Remove the clients that deduplicate their own data from the client deduplication list. See "Disabling MSDP client-side deduplication for a client" on page 107.
Step 2	Delete the storage units that use the disk pool	See the *NetBackup Administrator's Guide, Volume I*: https://support.cohesity.com/s/article/article-100040135
Step 3	Delete the disk pool	See "Deleting a Media Server Deduplication Pool" on page 550.
Step 4	Delete the deduplication storage server	See "Deleting an MSDP storage server" on page 538. Deleting the deduplication storage server does not alter the contents of the storage on physical disk. To protect against inadvertent data loss, NetBackup does not automatically delete the storage when you delete the storage server.
Step 5	Delete the configuration	Delete the deduplication configuration. See "Deleting the MSDP storage server configuration" on page 539.
Step 6	Delete the deduplication host configuration file	Each load balancing server contains a deduplication host configuration file. If you use load balancing servers, delete the deduplication host configuration file from those servers. See "Deleting an MSDP host configuration file" on page 200.
Step 7	Delete the storage directory and the database directory	Delete the storage directory and database directory. (Using a separate database directory was an option when you configured deduplication.) **Warning:** If you delete the storage directory and valid NetBackup images reside on the deduplication storage, data loss may occur. See the operating system documentation.

Chapter 14

Deduplication architecture

This chapter includes the following topics:

- MSDP server components
- Media server deduplication backup process
- MSDP client components
- MSDP client–side deduplication backup process

MSDP server components

Figure 14-1 is a diagram of the storage server components.

Figure 14-1 MSDP server components

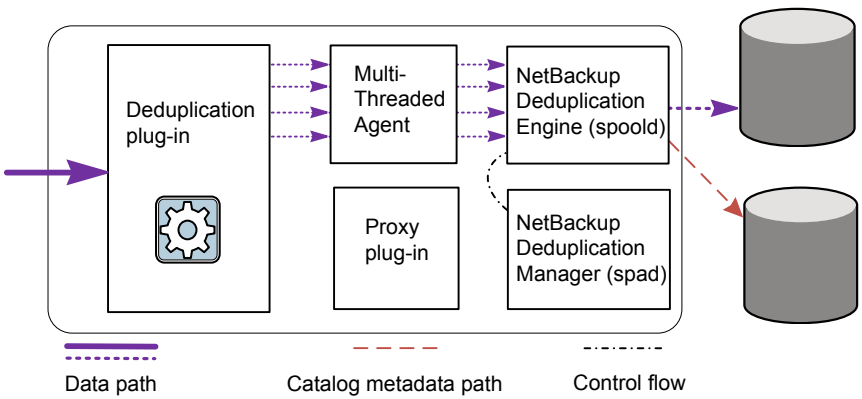


Table 14-1 describes the MSDP server components.

Table 14-1 NetBackup MSDP server components

Component	Description
Deduplication plug-in	The deduplication plug-in does the following: ■ Separates the file's metadata from the file's content. ■ Deduplicates the content (separates files into segments). ■ If required, compresses the data for backups and decompresses the backups for restores. ■ If required, encrypts the data for backups and decrypts the backups for restores. ■ If required, compresses the data for duplication and replication transfer. ■ If required, encrypts the data for duplication and replication transfer. The plug-in runs on the deduplication storage server and on load balancing servers.
Multi-Threaded Agent	The NetBackup Deduplication Multi-Threaded Agent uses multiple threads for asynchronous network I/O and CPU core calculations. The agent runs on the storage server, load balancing servers, and clients that deduplication their own data. See "About the MSDP Deduplication Multi-Threaded Agent" on page 66.
NetBackup Deduplication Engine	The NetBackup Deduplication Engine is one of the storage server core components. It provides many of the deduplication functions, which are described in Table 14-2. The binary file name is `spoold`, which is short for storage pool daemon; do not confuse it with a print spooler daemon. The `spoold` process appears as the NetBackup Deduplication Engine in the NetBackup web UI.
NetBackup Deduplication Manager	The deduplication manager is one of the storage server core components. The deduplication manager maintains the configuration and controls internal processes, optimized duplication, security, and event escalation. The deduplication manager binary file name is `spad`. The `spad` process appears as the NetBackup Deduplication Manager in the NetBackup web UI.
Proxy plug-in	The proxy plug-in manages control communication with the clients that back up their own data. It communicates with the OpenStorage proxy server (`nbostpxy`) on the client.
Reference database	The reference database stores the references that point to every data segment of which a file is composed. Unique fingerprints identify data segments. The reference database is partitioned into multiple small reference database files to improve scalability and performance. The reference database is separate from the NetBackup catalog. The NetBackup catalog maintains the usual NetBackup backup image information.

Table 14-2 describes the components and functions within the NetBackup Deduplication Engine.

Table 14-2 NetBackup Deduplication Engine components and functions

Component	Description
Connection and Task Manager	The Connection and Task Manager manages all of the connections from the load balancing servers and the clients that deduplicate their own data. The Connection and Task Manager is a set of functions and threads that does the following: ■ Provides a thread pool to serve all clients. ■ Maintains a task for each client connection. ■ Manages the mode of the Deduplication Engine based on the operation. Operations are backups, restores, queue processing, and so on.
Data integrity checking	The NetBackup Deduplication Engine checks the integrity of the data and resolves integrity problems. See "About MSDP data integrity checking" on page 554.
Data Store Manager	The Data Store Manager manages all of the data container files. The datastore Manager is a set of functions and threads that provides the following: ■ A transaction mechanism to back up data into the datastore. ■ A mechanism to read data from the datastore. ■ A transaction mechanism to reclaim space in the datastore (that is, compact containers and remove containers). Container IDs are unique. The Data Store Manager increments the container number with each new container created. The data in a container is never overwritten, and a container ID is never reused. See "About MSDP container files" on page 523.
Index Cache Manager	The Index Cache Manager manages the fingerprint cache. The cache improves fingerprint lookup speed. See "About the MSDP fingerprint cache" on page 73.
Queue processing	The NetBackup Deduplication Engine processes the transaction queue. See "About MSDP queue processing" on page 552.
Reference Database Engine	The Reference Database Engine stores the references that point to the data segments, such as read-from or write-to references. It manipulates a single database file at a time.

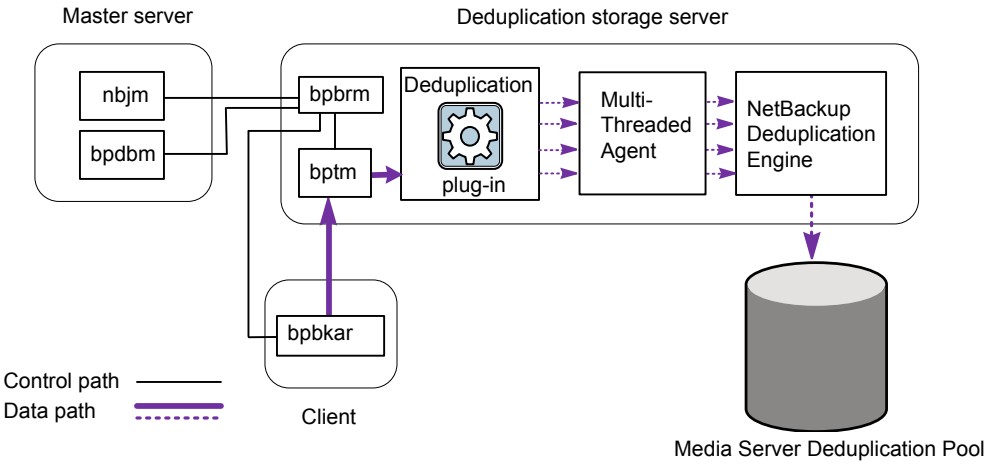
Table 14-2 NetBackup Deduplication Engine components and functions *(continued)*

Component	Description
Reference Database Manager	The Reference Database Manager manages all of the container references. It provides a transaction mechanism to manipulate a single database file.

Media server deduplication backup process

The Figure 14-2 diagram shows the backup process when a media server deduplicates the backups. The destination is a **Media Server Deduplication Pool**. A description follows.

Figure 14-2 Media server deduplication process



The following list describes the backup process when a media server deduplicates the backups and the destination is a **Media Server Deduplication Pool**:

- The NetBackup Job Manager (`nbjm`) starts the Backup/Restore Manager (`bpbrm`) on a media server.
- The Backup/Restore Manager starts the `bptm` process on the media server and the `bpbkar` process on the client.
- The Backup/Archive Manager (`bpbkar`) on the client generates the backup images and moves them to the media server `bptm` process.

The Backup/Archive Manager also sends the information about files within the image to the Backup/Restore Manager (`bpbrm`). The Backup/Restore Manager sends the file information to the `bpdbm` process on the primary server for the NetBackup database.

- The `bptm` process moves the data to the deduplication plug-in.
- The deduplication plug-in retrieves a list of IDs of the container files from the NetBackup Deduplication Engine. Those container files contain the fingerprints from the last full backup for the client. The list is used as a cache so the plug-in does not have to request each fingerprint from the engine.
- The deduplication plug-in separates the files in the backup image into segments.
- The deduplication plug-in buffers the segments and then sends batches of them to the Deduplication Multi-Threaded Agent. Multiple threads and shared memory are used for the data transfer.
- The NetBackup Deduplication Multi-Threaded Agent processes the data segments in parallel using multiple threads to improve throughput performance. The agent then sends only the unique data segments to the NetBackup Deduplication Engine.
  If the host is a load-balancing server, the Deduplication Engine is on a different host, the storage server.
- The NetBackup Deduplication Engine writes the data to the **Media Server Deduplication Pool**.
  The first backup may have a 0% deduplication rate, although a 0% rate is unlikely. Zero percent means that all file segments in the backup data are unique.

MSDP client components

Table 14-3 describes the client deduplication components.

Table 14-3 Client MSDP components

Component	Description
Deduplication plug-in	The deduplication plug-in does the following: ■ Separates the file's metadata from the file's content. ■ Deduplicates the content (separates files into segments). ■ If required, compresses the data for backups and decompresses the backups for restores. ■ If required, encrypts the data for backups and decrypts the backups for restores.

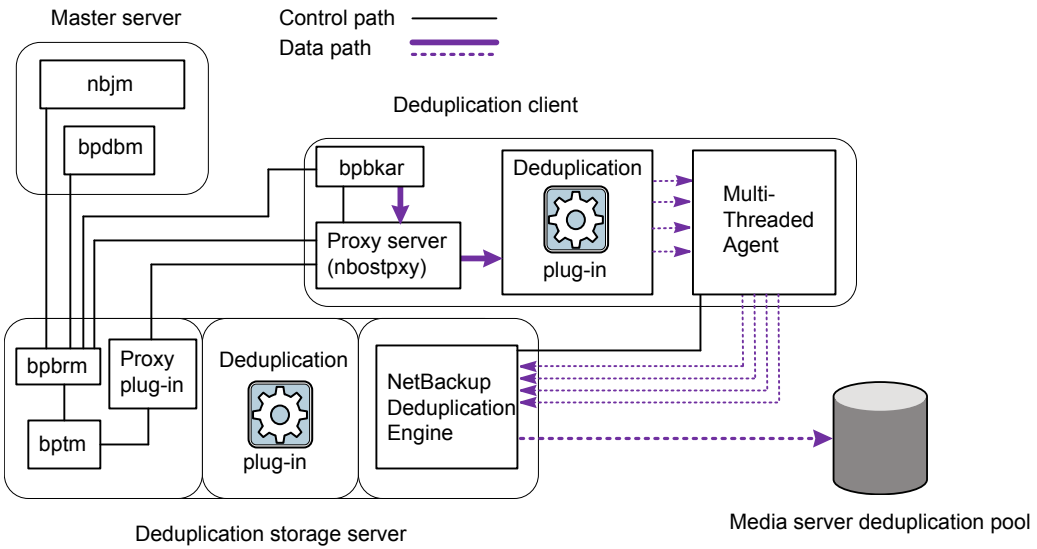
Table 14-3 Client MSDP components *(continued)*

Component	Description
Multi-Threaded Agent	The NetBackup Deduplication Multi-Threaded Agent uses multiple threads for asynchronous network I/O and CPU core calculations. The agent runs on the storage server, load balancing servers, and clients that deduplication their own data. See "About the MSDP Deduplication Multi-Threaded Agent" on page 66.
Proxy server	The OpenStorage proxy server (`nbostpxy`) manages control communication with the proxy plug-in on the storage server.

MSDP client–side deduplication backup process

The Figure 14-3 diagram shows the backup process of a client that deduplicates its own data. The destination is a media server deduplication pool. A description follows.

Figure 14-3 MSDP client backup to a deduplication pool



The following list describes the backup process for an MSDP client to a Media Server Deduplication Pool:

- The NetBackup Job Manager (`nbjm`) starts the Backup/Restore Manager (`bpbrm`) on a media server.

- The Backup/Restore Manager probes the client to determine if it is configured and ready for deduplication.
- If the client is ready, the Backup/Restore Manager starts the following processes: The OpenStorage proxy server (`nbostpxy`) on the client and the data moving processes (`bpbkar`) on the client and `bptm` on the media server).

  NetBackup uses the proxy plug-in on the media server to route control information from `bptm` to `nbostpxy`.
- The Backup/Archive Manager (`bpbkar`) generates the backup images and moves them to the client `nbostpxy` process by shared memory.

  The Backup/Archive Manager also sends the information about files within the image to the Backup/Restore Manager (`bpbrm`). The Backup/Restore Manager sends the file information to the `bpdbm` process on the primary server for the NetBackup database.
- The client `nbostpxy` process moves the data to the deduplication plug-in.
- The deduplication plug-in on the client tries to retrieve a list of fingerprints, in the following order:
  - From a client and a policy that is configured in the client's `pd.conf` file. The `FP_CACHE_CLIENT_POLICY` entry defines the client and policy to use for the fingerprint cache. The entry must be valid (that is, not expired).
    See "About seeding the MSDP fingerprint cache for remote client deduplication" on page 81.
  - From the previous backup for the client and policy.
  - From the special seeding directory on the storage server.
    See "About seeding the MSDP fingerprint cache for remote client deduplication" on page 81.

  The list of fingerprints is used as a cache so the plug-in does not have to request each fingerprint from the engine.
  If no fingerprints are loaded into the cache, the deduplication rate may be very low for the backup.
- The deduplication plug-in separates the files in the backup image into segments.
- The deduplication plug-in buffers the segments and then sends batches of them to the Deduplication Multi-Threaded Agent. Multiple threads and shared memory are used for the data transfer.
- The NetBackup Deduplication Multi-Threaded Agent processes the data segments in parallel using multiple threads to improve throughput performance. The agent then sends only the unique data segments to the NetBackupDeduplication Engine.

- The NetBackup Deduplication Engine writes the data to the **Media Server Deduplication Pool**.

  The first backup may have a 0% deduplication rate, although a 0% deduplication rate is unlikely. Zero percent means that all file segments in the backup data are unique.

Chapter 15

Configuring and managing universal shares

This chapter includes the following topics:

- Introduction to universal shares
- Prerequisites to configure universal shares
- Managing universal shares
- Mounting a universal share
- Creating a protection point for a universal share
- Restoring data using universal shares
- Advanced features of universal shares
- Managing universal share services
- Troubleshooting issues related to universal shares

Introduction to universal shares

This section provides an overview of universal shares, the features and key benefits of universal share, and how universal shares work.

Overview of universal shares

The universal share feature provides data ingest into an existing NetBackup deduplication pool (MSDP) or a supported Cohesity appliance using an NFS or a CIFS (SMB) share.

Both universal shares and MSDP use deduplication and compression.

Space efficiency is achieved by storing this data directly into an existing NetBackup-based Media Server Deduplication Pool.

Key benefits

The following information provides a brief description about the key benefits of universal shares:

- NAS-based storage target
  Unlike traditional NAS-based storage targets, universal shares offer all of the data protection and management capabilities that are provided by NetBackup, including Storage Lifecycle Policies (SLPs).
- Database dump location
  Universal shares offer a space saving (deduplicated) dump location, along with direct integration with NetBackup technologies including data retention, replication, and direct integration with cloud technologies.
- Financial and time savings
  Universal shares eliminate the need to purchase and maintain third-party intermediary storage. Use of this storage typically doubles the required I/O throughput since the data must be moved twice. Universal shares also cut in half the time it takes to protect valuable application or database data.
- Protection points
  The universal share protection point offers a fast point-in-time copy of all data that exists in the share. This copy of the data can be retained like any other data that is protected within NetBackup. All advanced NetBackup data management facilities such as Auto Image Replication (A.I.R.), storage lifecycle policies, optimized duplication, cloud, and tape are all available with any data in the universal share.
- Copy Data Management (CDM)
  The universal share protection point also offers powerful CDM tools. A read/write copy of any protection point can be "provisioned" or made available through a NAS (CIFS/NFS) based share. A provisioned copy of any protection point can be used for common CDM activities, including instant recovery or access of data in the provisioned protection point. For example, a database that has been previously dumped to the universal share can be run directly from the provisioned protection point.
- Back up and restore without client software and restore to any client
  Client software is not required for universal share backups or restores. You can also restore universal shares to any client. Universal shares work with any POSIX-compliant operating system that supports NFS or CIFS.

How to work with universal shares

The universal share feature provides a network-attached storage (NAS) option for supported Cohesity appliances as well as the software-only deployment of NetBackup. Traditional NAS offerings store data in conventional, non-deduplicated disk locations. Data in a universal share is placed on highly redundant storage in a space efficient, deduplicated state. The deduplication technology that is used for this repository is the same MSDP location used by standard client-based backups.

Any data that is stored in a universal share is automatically placed in the MSDP, where it is deduplicated automatically. This data is then deduplicated against all other data that was previously ingested into the media server's MSDP location. Since a typical MSDP location stores data across a broad scope of data types, the universal share offers significant deduplication efficiency. The protection point feature lets you create a point-in-time copy of the data that exists in the specified universal share. Once a protection point is created, NetBackup automatically catalogs the data as a specific point-in-time copy of that data and manages it like any other data that is ingested into NetBackup. Since the protection point only catalogs the universal share data that already resides in the MSDP, no data movement occurs. Therefore, the process of creating a protection point can be very fast.

Client support

The universal share feature supports a wide array of clients and data types. NetBackup software is not required on the client where the share is mounted. Any operating system that uses a POSIX-compliant file system and can mount a CIFS or an NFS network share can write data to a universal share. As the data comes in to the universal share, it is written directly into the Media Server Deduplication Pool (MSDP). No additional step or process of writing the data to a standard disk partition and then moving it to the deduplication pool is necessary.

Configuring and using an MSDP build-your-own (BYO) server for universal shares

Table 15-1 describes a high-level process for setting up an MSDP build-your-own (BYO) server for Universal Shares. (On an appliance, the universal share feature is ready to use as soon as storage is configured.) See the linked topics for more detailed information.

Table 15-1 Process for configuring and using universal shares with an MSDP build-your-own (BYO) server

Step	Description
1	Identify a machine. Make sure that the MSDP BYO server complies with prerequisites and hardware requirements. See "Build-your-own (BYO) server prerequisites and hardware requirements to configure universal shares" on page 604..
2	In the NetBackup web UI, create a universal share. See *Create a universal share* in the NetBackup Web UI Administrator's Guide.
3	Mount the universal share that was created from the NetBackup web UI. See "Mounting a universal share" on page 628.
4	Configure a universal share backup policy.
5	Optionally, use the ingest mode to dump data or to load backup data from a workload to the universal share over NFS/CIFS. When ingest mode is turned on, the backup script triggers the universal share to persist all the data from memory to disk on the client side at the end of the backup or the dump. Ingest mode is faster than normal mode as it does not guarantee all the ingest data is persisted to disk until the ingest mode is turn off. See "Ingest data to a universal share with the ingest mode" on page 654.
6	Restore from a universal share backup.

Prerequisites to configure universal shares

There are prerequisites and hardware requirements to configure universal shares. After the requirements are completed, you can configure and manage an MSDP build-your-own (BYO) server for universal shares.

Build-your-own (BYO) server prerequisites and hardware requirements to configure universal shares

The following are prerequisites for using the universal share MSDP build-your-own (BYO) server feature:

- The universal share feature is supported on an MSDP BYO storage server with Red Hat Enterprise Linux 7.6 or later.
- The universal share feature is not supported on SUSE Linux.
- You must set up user authentication for the universal share.

See "Configuring universal share user authentication" on page 607.

- NFS services must be installed and running if you want to use the share over NFS.
  Ensure that the `nfs-utils` is installed:
  - Red Hat: `yum install nfs-utils -y`
- Samba services must be installed and running if you want to use share over CIFS/SMB.
  Ensure that the Linux `samba` and `samba winbind` packages are installed.
  - `yum install samba samba-common samba-winbind samba-winbind-clients samba-winbind-modules -y`
- You must configure Samba users on the corresponding storage server and enter the credentials on the client.
  See "Configuring universal share user authentication" on page 607.
- Ensure that the `xfsprogs` is installed:
  - Red Hat: `yum install xfsprogs -y`
- Ensure that the following commands are run to grant permissions to the SMB shares:
  - `setsebool -P samba_export_all_rw on`
  - `setsebool -P samba_export_all_ro on`
- Install and run NGINX. The minimum recommended NGINX version is 1.24.0.
- After NGINX is installed, the HTTP web service at port 80 is enabled by default. Remove `/etc/nginx/conf.d/default.conf` or edit the file to disable the HTTP web service if it is not needed.
- Ensure that the `/mnt` folder on the storage server is not directly mounted by any mount points. Mount points should be mounted to its subfolders.
- Ensure that you configure the storage server for MSDP. If you configure the universal share feature on BYO after storage is configured or upgraded without the NGINX service installed, run the command on the storage server:
  `/usr/openv/pdde/vpfs/bin/vpfs_config.sh --configure_byo`
- Ensure that the required network ports are open.
  See "NetBackup media server ports" in the *NetBackup Network Ports Reference Guide*.

Table 15-2 Hardware configuration requirements for universal shares on a Build Your Own (BYO) server

CPU	Memory	Disk
■ Minimum 2.2-GHz clock rate. ■ 64-bit processor. ■ Minimum 4 cores; 8 cores recommended. For 64 TBs of storage, the Intel x86-64 architecture requires eight cores. ■ Enable the VT-X option in the CPU configuration.	■ 16 GB (For 8 TBs to 32 TBs of storage - 1GB RAM for 1TB of storage). ■ 32 GBs of RAM for more than 32 TBs of storage. ■ An additional 500MB of RAM for each live mount.	Disk size depends on the size of your backup. Refer to the hardware requirements for NetBackup and Media Server Deduplication Pool (MSDP). If a system has multiple data partitions, all the partitions must be the same size. Example: If a BYO server has a first partition at 4TB, all additional data partitions must be at 4TB in size.

NetBackup upgrades

You must unmount all the NFS mount points on the client side before you upgrade to NetBackup to avoid issues when accessing the universal share on the client side over NFS.

Note: The CIFS/SMB shares do not require these operations.

1. Unmount all the universal share that were mounted on the Linux UNIX client.
2. Upgrade to NetBackup.
3. Start the NetBackup services.
4. Mount the universal share again on the Linux UNIX client.

Managing the deduplication web service user and the user group for MSDP BYO server

When an MSDP is configured, the user `spws` and the group `spwsgrp` are created and used by Storage Platform Web Service (SPWS) service by default.

If you do not want to use the default user and group for the SPWS service, you can configure the SPWS service before or after you configure the MSDP.

To manage the SPWS service before MSDP server is configured

- If the MSDP storage server is not configured yet, run the following command on the storage server before you configure an MSDP.

  ```
  /usr/openv/pdde/vpfs/bin/spws_config.sh --spwsuser=<spwsuser>
  --spwsgroup=<spwsgroup>
  ```

  When you configure the MSDP, SPWS service will use the specified user and group and `spws` and the group `spwsgrp` will not be created.

To manage the SPWS service after the MSDP server is configured

- If the MSDP storage server is already configured, and the user `spws` and the group `spwsgrp` are created, run the following command on the storage server to specify the new user and the group.

  ```
  /usr/openv/pdde/vpfs/bin/spws_config.sh --spwsuser=<spwsuser>
  --spwsgroup=<spwsgroup>
  ```

  The SPWS service changes to use the specified user and the group. The owner of the files that are used by the SPWS service is also changed.

  It does not delete the user `spws` and the group `spwsgrp` that are created. You can delete the user and the group manually.

Note: You must run the `spws_config.sh` script under the root user so that it can create the log file in `/var/run/vpfs/` and configure the system services.

Configuring universal share user authentication

The universal share created with CIFS/SMB protocol supports three methods of user authentication:

- Active Directory-based user authentication
  See "Active Directory-based authentication" on page 607.
- Local user-based authentication
  See "Local user-based authentication" on page 610.
- Kerberos-based authentication
  See "Kerberos-based authentication" on page 611.

Active Directory-based authentication

If the appliance, Flex Scale, Flex Appliance application instance, or MSDP BYO server is part of the Active Directory domain, you can use this approach.

When you create a universal share from the NetBackup web UI, you can specify Active Directory users or groups. This approach restricts access to only specified users or groups. You can also control permissions from the Windows client where the universal share is mounted. See the NetBackup Web UI Administrator's Guide for more information.

For information about setting up Active Directory users or groups with an appliance, see the NetBackup Appliance Security Guide.

Universal shares can be created with NFS or SMB protocol. When the SMB protocol is used, SMB must be set up with ADS or in local user mode. The following table describes how to configure the media server with Active Directory for various platforms and create a universal share using SMB.

Table 15-3 Describes the requirements for different platforms to join the Active Directory domain

Platform	Requirements
BYO appliance	For BYO, `/usr/openv/pdde/vpfs/bin/register_samba_to_ad.sh` is used for joining an Active Directory domain. Example usage of `register_samba_to_ad.sh`: `/usr/openv/pdde/vpfs/bin/register_samba_to_ad.sh --domain=<domain> --username=<username>` The following are other options you can use with `register_samba_to_ad.sh`: `--domain=<domain> : domain name` `--domaincontroller=<domain controller> : domain controller` `--username=<username> : windows domain username which has the privilege to join the client to domain` `--help|-h : Print the usage`
NetBackup appliance (NBA)	Review the section *Adding an Active Directory server configuration* in the NetBackup Appliance Administrator's Guide.
Flex media server	The same as BYO.

Table 15-3 Describes the requirements for different platforms to join the Active Directory domain *(continued)*

Platform	Requirements
WORM enabled storage server	The storage server can be configured to join or leave Active Directory with Restricted Shell commands. `[msdp-16.0] hostname >` `setting ActiveDirectory` `configure ad_server=<ad_server>` `domain=<domain_server>` `domain_admin=<domain_adin>` See "Connecting an Active Directory domain to a WORM or an MSDP storage server for Universal Shares and Instant Access" on page 713.
Flex Scale	Review the section *Configuring AD server for Universal shares and Instant Access* in the NetBackup Flex Scale Administrator's Guide.
AKS/EKS AD	NetBackup support only SMB local user mode. The SMB server is configured with local user mode by default.

Once the storage server has been added to an Active Directory domain, a universal share can be created as normal. Any users and user groups that are specified are validated using the `wbinfo` command. The following procedure describes how to add a universal share to an Active Directory.

Adding a universal share to an Active Directory

1. Open the NetBackup web UI.
2. Create a universal share with the SMB protocol.
3. Mount the shared storage on a Windows client.

   Provide all necessary credentials.
4. Verify that the universal share is fully set up, and can be backed up and restored using a **Universal-Share** policy.

The following requirements exist to add Microsoft SQL Server instant access to an Active Directory:

- The storage server and the client must be in the same domain.
- The domain user account is required with the necessary permissions to log on to the Microsoft SQL Server client.
- In the web UI, register the Microsoft SQL Server instance with the domain user.

- See the information on manually adding a SQL Server instance in the *NetBackup for Microsoft SQL Server Administrator's Guide*.
- The domain user credentials are required to use instant access.

Local user-based authentication

You must configure SMB users on the corresponding storage server and enter the credentials on the client.

If the SMB service is part of a Windows domain, the Windows domain users can use the SMB share. In this scenario, credentials are not required to access the share.

Note: For Azure Kubernetes Service (AKS) and Amazon Elastic Kubernetes Service (EKS) cloud platforms, only a SMB local user can access the SMB share. You must add SMB users to access the SMB share.

If the SMB service is not part of Windows domain, perform the following steps:

- For a NetBackup Appliance:
  For a NetBackup Appliance, local users are also SMB users. To manage local users, log in to the CLISH and select **Main > Settings > Security > Authentication > LocalUser**. The SMB password is the same as the local user's login password.
- For an MDSP BYO server:
  For an MDSP BYO server, create a Linux user (if one does not exist). Then, add the user to SMB.
  For example, the following commands create a username for the SMB service only:
  ```
  adduser --no-create-home -s /sbin/nologin <username>
  smbpasswd -a <username>
  ```
  To add an existing user to the SMB service, run the following command:
  ```
  smbpasswd -a username
  ```
- For a Flex Appliance primary or media server application instance:
  For a Flex Appliance primary or media server application instance, log in to the instance and add any local user to the SMB service as follows:
  - If desired, create a new local user with the following commands:
    ```
    useradd <username>
    passwd <username>
    ```
    You can also use an existing local user.

- Run the following commands to create user credentials for the SMB service and enable the user:

  ```
  smbpasswd -a <username>
  smbpasswd -e <username>
  ```

- For a WORM storage server application instance:
  For a WORM storage server instance, log in to the instance and add a local SMB user with the following command: `setting smb add-user username=<username> password=<password>`
  You can view the new user with the `setting smb list-users` command. To remove a user, run the `setting smb remove-user username=<username>` command.
- For the AKS and the EKS cloud platform:
  - Log in to the MSDP engine pod in a cluster using `kubectl`.
  - Run the following command to log in to Restricted Shell in the MSDP engine.
    `su - msdpadm`
  - Run the following Restricted Shell command to add a SMB user.
    `setting smb add-user username=<username>`
    For example,
    `msdp-16.1] > setting smb add-user username=<username>`
    You can use the same command to update the password for an existing user.
    In AKS and EKS cloud platforms, the SMB Restricted Shell command configures SMB servers in all MSDP engines in a cluster.

Kerberos-based authentication

Use Kerberos-based authentication for universal shares to secure the connection between clients and servers. All the Kerberos security types krb5, krb5i, and krb5p are supported for the universal share configuration.

You must follow these steps to configure the Kerberos authentication for universal shares.

Table 15-4

Step	Task	Description
1.	Configure the Active Directory-based authentication.	For information on how to add storage servers to the Active Directory domain, See "Active Directory-based authentication" on page 607.
2.	On Windows Active Directory domain server, create Active Directory users for Kerberos authentication.	See "Creating Active Directory users for Kerberos authentication " on page 612.
3.	Register Kerberos principals to KDC database.	See "Registering the Kerberos principals to the KDC database" on page 613.
4.	Configure Kerberos-based authentication on the servers.	See "Configuring the Kerberos-based authentication on the servers and the clients" on page 613.

Creating Active Directory users for Kerberos authentication

After storage servers are added to the Active Directory domain, perform the following tasks before you configure Kerberos-based authentication for universal shares on the NetBackup web UI.

- On Windows Active Directory domain server, create Active Directory users for Kerberos authentication.
- Register Kerberos principals to KDC (Key Distribution Center) database. See "Registering the Kerberos principals to the KDC database" on page 613.

To create Active Directory users for Kerberos authentication.

1. Log in to the Windows Active Directory domain server.
2. Navigate to **Start > Administrative Tools > Active Directory Users and Computers**.
3. In the left pane, select the correct domain name and then select **Users**.
4. Right-click **Users** and select **New** > **User**.

5 Enter the domain user information. **User logon name** is used for Active Directory domain login and authentication.

For storage servers, the logon name must be *nfs/<storage server FQDN>*. Where the *nfs* is an NFS service principal and *storage server* is the host where your universal shares are created. For example, *nfs/storage-server.mydomain.com*.

For a universal share server, create one more user *host/<storage server FQDN>*.

For a universal share server, you must create two Active Directory users, *nfs/<storage server FQDN>* and *host/<storage server FQDN>*. For a universal share client, create only one user, *host/<universal share client FQDN>*.

6 Set password for the new user.

7 Click **Finish** to finish the user creation.

8 Double-click the user you have created to open the property window.

9 In **Account options** list, select **AES 128** and **AES 256** encryption items.

Registering the Kerberos principals to the KDC database

After Active Directory users for Kerberos authentication are created, register the Kerberos principals to the KDC database.

To register Kerberos principals to the KDC database.

- Open the command-line and run `ktpass` command to register the principal to KDC database.

  For example,

  ```
  ktpass -princ nfs/storage-server.mydomain.com@MYDOMAIN.COM
  -mapuser MYDOMAIN\username -pass <password> -ptype
  KRB5_NT_PRINCIPAL -crypto All -out storage-server.keytab
  ```

  Where `MYDOMAIN\username` is the **User logon name (pre-Windows 2000)** in the user property page.

  Note: Password must be the password of the Active Directory user. Otherwise, the previous password will be modified.

Configuring the Kerberos-based authentication on the servers and the clients

You can configure the Kerberos-based authentication for NetBackup BYO, Flex media server, Flex WORM, and Flex Scale.

You must configure Kerberos-based authentication both on the servers and the clients.

For NetBackup BYO environment, before you configure Kerberos authentication on NetBackup servers and clients, check if the necessary krb5 package is installed on the system. Run the following commands to check if these packages are installed or not:

```
yum install krb5-workstation
```

```
pam_krb5 -f
```

To configure Kerberos-based authentication on the servers

- On the NetBackup server, run the `vpfs_nfs_krb.sh` script to create `keytab` entries for Kerberos principals.

  ```
  /usr/openv/pdde/vpfs/bin/vpfs_nfs_krb.sh
  ```

  For NetBackup BYO, run the script in the command window. For Flex media server, you must log in to the media server instance and run the script with `sudo`.

  - Add the key entries.

    ```
    ./vpfs_nfs_krb.sh add --user nfs/storage-server.mydomain.com
    ```

  - Delete the key entries.

    ```
    ./vpfs_nfs_krb.sh delete --user nfs/storage-server.mydomain.com
    ```

  - Verify Kerberos principal login.

    ```
    ./vpfs_nfs_krb.sh verify --user nfs/storage-server.mydomain.com
    ```

  - Update the password for Kerberos principals.

    ```
    ./vpfs_nfs_krb.sh update --user nfs/storage-server.mydomain.com
    ```

  - Display the key entries.

    ```
    ./vpfs_nfs_krb.sh list
    ```

  - Display the configurations related to Kerberos authentication.

    ```
    ./vpfs_nfs_krb.sh status
    ```

  For Flex WORM and Flex Scale, you must log in to the WORM or MSDP engine Restricted Shell to run these commands.

  - Add the key entries.

    ```
    setting SecureNfs add-krb-user
    krbuser=nfs/storage-server.mydomain.com
    ```

  - Delete the key entries.

    ```
    setting SecureNfs delete-krb-user
    krbuser=nfs/storage-server.mydomain.com
    ```

- Verify Kerberos principal login.

  ```
  setting SecureNfs verify-krb-user
  krbuser=nfs/storage-server.mydomain.com
  ```

- Update the password for Kerberos principals.

  ```
  setting SecureNfs update-krb-user
  krbuser=nfs/storage-server.mydomain.com
  ```

- Display the key entries.

  ```
  setting SecureNfs list-krb-users
  ```

- Display the configurations related to Kerberos authentication.

  ```
  setting SecureNfs nfs-secure-status
  ```

Both *nfs/storage-server.mydomain.com* and *host/storage-server.mydomain.com* principals must be added to the `/etc/krb5.keytab` in the storage servers.

For Flex Scale, you must create both *nfs/storage-server.mydomain.com* and *host/storage-server.mydomain.com* principals for every MSDP engine. Here, the *storage-server* is the MSDP engine host name configured in Flex Scale web UI. You can find these names in **Monitor > NetBackup > Storage servers list** on the NetBackup web UI. All these principals must be added to the `krb5.keytab` file by running the MSDP shell command. In every engine, the `/etc/krb5.keytab` file contains key entries of all principals that are created for all engines in the cluster.

For multi-VLAN environments, storage servers may have more than one IPs. If you need to mount the universal shares from the clients that are in the secondary VLAN, ensure that other FQDNs of the storage servers and clients are added in DNS, and corresponding Active Directory users are created and registered as Kerberos principals. The key entries also need to be added to the `/etc/krb5.keytab` file.

To configure Kerberos-based authentication on the universal share clients

1 Create `/etc/krb5.conf` file for the Kerberos authentication.

You can copy the `/etc/krb5.conf` file from a storage server where universal share is configured.

Note: If there is `kdc` section defined in `krb5.conf` file. Copy `kdc.conf` file along with `/etc/krb5.conf` file.

---

2 Enable `SECURE_NFS` in the `/etc/sysconfig/nfs` file.

Add the line `SECURE_NFS=yes` in the `/etc/sysconfig/nfs` configuration file.

Then, run the following command to restart the service:

```
systemctl restart nfs-secure
```

Note: This configuration is required only on Red Hat 7 or earlier versions. On Red Hat 8 and 9, this step is not required.

---

3 Create keytab entries for Kerberos principals.

You can configure the keytab file by using one of the following two methods:

- Copy `vpfs_nfs_krb.sh` script from a storage server, then run the script to configure the keytab file.
- After the Active Directory user for a universal share client is created, run `ktpass` utility to generate the keytab for the Kerberos principal.
  Then, copy the keytab file to the NFS client `/etc` folder and rename it to `/etc/krb5.keytab`.

Note: If the universal share client has the existing `/etc/krb5.keytab` file, use the `vpfs_nfs_krb.sh` script to add the key entries.

The script `vpfs_nfs_krb.sh` can write logs about universal share configuration-related operations. The logs are available only for universal share servers.

You can find the logs at the following location: `/<storage path>/log/vpfs/yymmdd_*_vpfs_nfs_krb.log`

Managing universal shares

You can manage universal shares across NetBackup Appliance, Flex Appliance, Flex Scale, Flex WORM/non-WORM, MSDP AKS/EKS deployment, build-your-own (BYO) and BYO-In-Cloud servers. You can create, view, edit, or delete a universal share.

Create a universal share

A universal share offers the ability to ingest data directly into a space efficient SMB (CIFS) or NFS share. Space efficiency is achieved by storing the ingested data directly to an existing NetBackup deduplication pool (MSDP). No NetBackup software needs to be installed on the client that mounts the share. Any operating system that is running a POSIX-compliant file system and can mount an SMB (CIFS) or NFS network share can write data to a universal share.

If you want to create a universal share with object store you have to first create a storage server and then create a cloud volume. When you create a universal share, select the cloud volume that you created.

If you want to view specific storage servers containing universal shares, select **Select storage server** in the top right. Then, select the storage servers that contain universal shares, and they are displayed in the table.

To create a universal share in the NetBackup web UI

1. If necessary, configure an MSDP storage server.

   See “Create a Media Server Deduplication Pool storage server ” on page 296.

2. On the left, select **Storage > Disk storage**.
3. Select the **Universal shares** tab. Then select **Add**.
4. Provide the following required information:
   - Enter a **Display name**. This name is used in the universal share path.
   - Select a **Type**. If you want to set up **Cloud cache properties**, you must select **Regular**. If **Accelerator** type is selected, you must specify the **Disk volume**.
     If you want to direct data to a local disk storage or direct data to cloud, select the type as **Cluster**.
   - Select the **Storage server**.
   - Select the **Disk volume**.
     When **Accelerator** is selected in **Type**, you can only select a cloud disk volume that is configured on the MSDP storage server.

Select the search icon to get the volume list, and select the disk volume. **PureDiskVolume** is selected by default.

This option is available only if universal share with object storage in cloud feature is enabled.

- In **Cloud cache properties**, specify the size of the local disk cache in the **Request cloud cache disk space**.

  The **Request cloud cache disk space** can only be set here on initial setup. Any subsequent changes must be made on the storage server properties page.

  Note: When you update the **Cloud cache properties** setting in the storage server properties page, there is an interruption of the current shared mounts. When you select **Save**, the `vpfsd` process restarts to apply the new value.

  In addition, new universal shares cannot be created if the available size is less than 128GB.

- Select the **Protocol**: **NFS** or **SMB** (CIFS)

  Select the Direct NFS check box to improve the performance of Network Attached Storage (NAS) over standard NFS for Oracle Databases. With direct NFS, the **insecure** export option is added to the export path on the NFS server.

  For example, `/mnt/vpfs_shares/dnfs/dnfs client-abcd(rw,insecure,mp,fsid=4161bb04-f62f-40f3-af09-0d9a8713694b)`

  See "Use Direct Network File System to improve the performance of Network Attached Storage" on page 622.

- If you select the type as **Cluster**, select the exported storage server.

  The exported storage server is the server on which the universal share export path is available for export to the clients.

  You can also select multiple servers.

- Specify a **Host** that is allowed to mount the share and then select **Add to list**. You can use the host name, IP address or range, short name, or the FQDN to specify the **Host**. You can enter multiple hosts for each share.

  NetBackup uses a set of rules to validate any wildcard characters that are used for the **Host** that you specify. More information about **Host** validation is available:

  See "About Host wildcard validation for universal share creation" on page 620.

When **Accelerator** is selected in **Type**, the **Host** can only be FQDN.

5 At this point, continue to enter values in the remaining fields or select **Save** to save the universal share. You can update the remaining fields later from the universal share's details page:

- Select a **Quota** type: **Unlimited** or **Custom**. If you select **Custom**, also specify the quota in MB, GB, or TB units.
  The **Custom** quota value limits the amount of data that is ingested into the share. Quotas are enforced using the front-end terabyte (FETB) calculation method. They are Implemented per share and can be modified at any time. You do not need to remount the share for the change to a take effect.
  To update the quote type or value from the universal share's details page, select **Edit** in the Quota section.
- Specify the **User names** (Local or Active Directory) and the **Group names** (Active Directory only). Only the specified users or groups can access the share. You can add and update the **User names** and the **Group Names** later from the details page of an existing universal share.

  Note: Currently, the **User names** and the **Group names** are supported only for the SMB (CIFS) protocol.

- Specify Kerberos security methods if the selected protocol is NFS and the Kerberos service is supported on the selected Storage server.
  If you select more than one Kerberos security methods, you can specify any method as a mount command option to the share from the client host.
  - Kerberos 5
    Uses Kerberos V5 instead of local UNIX UIDs and GIDs to authenticate the users.
  - Kerberos 5i
    Uses Kerberos V5 for user authentication and performs integrity checking of NFS operations using the secure checksums to prevent tampering of the data.
  - Kerberos 5p
    Uses Kerberos V5 for user authentication and integrity checking. It encrypts NFS traffic to prevent traffic sniffing. This option is the most secure setting but it also involves the most performance overhead.

Note: The image sharing storage server is not available when you create a new universal share.

About Host wildcard validation for universal share creation

NetBackup uses the following set of rules to validate wildcard characters that are used in the **Host** that you specify when you create a universal share.

Host name or fully qualified domain name (FQDN) validation rules:

- Allow a single host name or FQDN.
- Allow '*' and '?' wildcards only for FQDNs.
  Do not allow any other wildcard.
- Validate the host name or FQDN validated according to the RFC standard. Refer to this resource for more information:
  https://en.wikipedia.org/wiki/Hostname

IP address or IP Network/Range validation rules:

- Allow a single host where one particular host is specified with an IP address.
- Allow '/' only with IP address range. For example: 192.168.0.0/28. allows the first 16 IP addresses, from 192.168.0.0 to 192.168.0.15, to access the exported file system, but not 192.168.0.16 and higher.
  Do not allow any other wildcard or special character.
- Validate IP address, IP range, and FQDNs using regex or library functions.

The following examples are valid **Host** entries that use wildcards:

```
*.example.com

..example.com

*.vxindia.veritas.com

*.veritas.com

some.example.com

*.some.example.com

s???me.example.com

s?me.example.com

*.example.com

*.veritas.com
```

```
*.example.com

..example.com

*.vxindia.veritas.com

*.veritas.com
```

The following examples are not valid:

```
so*me.example.com/

s?me.examp!e.com/

s*me.examp!!!!e.com

so*me.example.com/

s?me.examp!e.com/

s*me.examp!!!!e.com

some.example.com?

some.example.com*

some.example.com?

some.example.com*

some.ex*ample.com

s*ome.example.com

s*me.example.com

some.example.com?

some.example.com*

some.ex*ample.com

s*ome.example.com
```

```
s*me.example.com

*some.example.com
```

Using instant access for MS-Windows, Standard, and Universal-Share policies

Instant access for unstructured data assets allows users to create instant access mounts from the backup images that are created with MS-Windows, Standard, or Universal-Share policies.

To manage instant access with a MS-Windows, Standard, or Universal-Share policy, a user must have the RBAC Administrator role. Or, a role with similar permissions.

The backup images that are created using the MS-Windows policy and mounted over instant access, display the file ownership and access control list (ACL).

When you create an instant access livemount for an NTFS filesystem from a MS-Windows backup, use the 'shareReader' parameter in the json file (instant access live mount creation API) to view the original file security information (file ACL and file ownership).

Format: `shareReader="Domain\\reader"`

The username specified for shareReader must be a valid AD (active directory) user. For an instant access livemount created with the 'shareReader' parameter, all the AD users with permissions to access a share file in the ACL and AD user 'shareReader' can access it. If you do not want the livemount to display the original backup file security information, avoid using the 'shareReader' parameter.

You can instantly access backup copies from a local or a cloud LSU (logical storage unit) using NetBackup Instant Access APIs.

You can instantly access backup copies of universal shares from the NetBackup web UI or using the NetBackup Instant Access APIs.

MS-Windows and Standard policies only support instant access using REST APIs.

Note: Instant access requires the following services: NGINX, NFS. SAMBA, WINBIND (if Active Directory is required), SPWS, VPFS.

Use Direct Network File System to improve the performance of Network Attached Storage

Direct Network File System (dNFS) provides performance improvements for Network Attached Storage (NAS) over standard NFS for Oracle Databases. Direct NFS allows Oracle software to skip the operating system's NFS client when it

communicates with a storage server. Direct NFS also improves High Availability (HA) and scalability by supporting up to four parallel network paths to storage and load-balancing across these paths. These improvements result in cost savings for database storage.

Direct NFS Requirements

- NFS servers must have write size values (wtmax) of 32768 or higher.
- NFS mount points must be mounted both by operating system NFS client and Direct NFS client.
- Set the NFS buffer size parameters rsize and wsize to at least 1048576 using the following command:

  ```
  rsize and wsize
  nfs_server:/vol/DATA/oradata /mnt/ nfs\
  rw,bg,hard,nointr,rsize=1048576,wsize=1048576,tcp,actimeo=0,vers=3,timeo=600
  ```

- Ensure that the TCP network buffer size is large enough to not hinder Direct NFS performance. The following command can verify the TCP buffer size:

  ```
  sysctl -a |grep -e net.ipv4.tcp_[rw]mem
  ```

  TCP buffer output

  ```
  net.ipv4.tcp_rmem = 4096 87380 1056768
  net.ipv4.tcp_wmem = 4096 16384 1056768
  ```

  To change the buffer size, open `/etc/sysctl.conf` as root and modify the following values:

  sysctl.conf

  ```
  net.ipv4.tcp_rmem = 4096 87380 4194304
  net.ipv4.tcp_wmem = 4096 16384 4194304
  ```

  Before running `sysctl -p`, restart the network with `/etc/rc.d/init.d/network restart`.

Enable and disable Direct NFS

To enable Direct NFS, run the following commands and restart the database instance:

```
cd $ORACLE_HOME/rdbms/lib

make -f ins_rdbms.mk dnfs_on
```

To enable Direct NFS, run the following commands and remove the `oranfstab` file:

```
cd $ORACLE_HOME/rdbms/lib

make -f ins_rdbms.mk dnfs_off
```

Direct NFS client configuration

In the following directories of Direct NFS, search for the `oranfstab` file where the first matching entry (of the file) is the mount point. You can update the file to set up multipathing and handle other configuration details.

- $ORACLE_HOME/dbs
- /var/opt/oracle
- /etc/mnttab

Use the following list of parameters to create the `oranfstab` file for each NFS server that you want to access using Direct NFS:

Table 15-5 Parameters to create the oranfstab file

Parameter	Usage
Server	Unique identifier for the NFS server.
Local	Network paths (up to 4) on the database host.
Path	Network paths (up to 4) on the NFS server.
Export	The exported volume on the NFS server.
Mount	The local mount point for the exported volume.
mnt_timeout	Time in seconds to wait for the first mount.
dontroute	The operating system routing of outgoing messages is prevented.
management	Network path for NFS server management interface.
nfs_version	The NFS protocol version that the Direct NFS client uses.
security_default	The default security mode that is applicable for all the exported NFS server paths for a server entry.
security	The security level to enable security with the Kerberos authentication protocol with Direct NFS client.
community	The community string for use in SNMP queries.

Sample output of an `oranfstab` file.

```
server:  myNFSServer1
local: 192.168.1.1 path: 192.168.1.2
local: 192.168.2.1 path: 192.168.2.2
local: 192.168.3.1 path: 192.168.3.2
```

```
local: 192.168.4.1 path: 192.168.4.2
export: /vol/oradata1 mount: /mnt/oradata1
export: /vol/oradata2 mount: /mnt/oradata2
mnt_timeout: 600
```

Ensure that you set up the `oradism` file at the following path: `$ORACLE_HOME/bin/oradism`. Direct NFS uses this `oradism` binary to issue mounts as root. The file must be local to each node and with the ownership of a root user.

To ensure that the file is local to each node, run the `chown root $ORACLE_HOME/bin/oradism` command. Run `chmod 4755 $ORACLE_HOME/bin/oradism` to ensure that the `oradism` file has the correct access permissions.

Client monitoring

Refer to the contents of the following tables for client monitoring.

Table 15-6 v$ tables

Item	Description
`v$dnfs_servers`	Lists the NFS servers that the Direct NFS client has mounted.
`v$dnfs_files`	Lists the files that the Direct NFS client has opened.
`v$dnfs_channels`	Lists the TCP connections that are established from the NFS server to Direct NFS.
`v$dnfs_stats`	Lists the statistics on the different NFS operations that the Oracle processes have issued.

Configure Direct NFS on Windows Platform

Ensure that Oracle 11g software or later is installed using the Oracle installer on the Windows server.

Create and configure the `oranfstab` file. You must add the `oranfstab` file in the `%ORACLE_HOME%\dbs` directory. Ensure that any file extension (for example, text file - txt) is added in the file name.

Configure the `oranfstab` as in the following way:

```
C:\>type %ORACLE_HOME%\dbs\oranfstab
server: lnxnfs          <=== NFS server Host name
```

```
path: 10.171.52.54 <--- First path to NFS server ie NFS server NIC
local: 10.171.52.33 <--- First client-side NIC
export: /oraclenfs  mount: y:\
uid:1000
gid:1000
C:\>
```

The Direct NFS client uses the UID or the GID value to access all NFS servers that are listed in the `oranfstab` file. Direct NFS ignores a UID or the GID value of 0. The UID and the GID used in the earlier example is of an Oracle user from the NFS server.

The exported path from the NFS server must be accessible for read, write, and run operations by the Oracle user with the UID and the GID specified in the `oranfstab` file. If neither UID nor GID is listed, the default value of 65534, is used to access all NFS servers listed in `oranfstab` file.

View or edit a universal share

You can view the details of a universal share or edit certain attributes of the universal share.

View details of a universal share

To view the details of a universal share

1. On the left, select **Storage > Disk storage**. Then select the **Universal shares** tab.
2. Locate the universal share and click on its name.

   Use the **Filters** to display specific universal shares. For example, universal shares with the **SMB** protocol or universal shares whose state is **Exported**.

   The **ID** is the UUID of the universal share.

   The **Export path** is the path is used in a universal share backup policy.

   The **Mount path** is the path that is used to connect from the client.

Edit a universal share

You can edit the quota for the share and the hosts that can mount the share.

To edit a universal share

1. On the left, select **Storage > Disk storage**. Then select the **Universal shares** tab.
2. Locate the universal share and click on its name.
3. You can edit the following details for the universal share.

Quota	Select **Edit** to change the quota for the share.
Exported storage servers	Select **Edit** to change the NFS or the SMB export servers of the universal share.
Hosts	Select **Edit** to add or delete the hosts that can mount the share.
Kerberos	Select **Edit** to change the Kerberos security method if the selected protocol is NFS and the Kerberos service is supported on the selected storage server. See "Kerberos-based authentication" on page 611. **Note:** When the Kerberos security method is updated, it affects the ability to connect NFS server from the clients that are configured in the current universal share. Use the Kerberos security method you have updated as a mount command parameter to mount the NFS server again.

Delete a universal share

You can delete a universal share from NetBackup storage.

Deleting a universal share also deletes all data in the share. This action is irreversible and may take some time if the amount of data is large. Any active data transfers are immediately terminated, any mounted shares are immediately removed and become inaccessible on the clients. Unmount the universal shares on the client before removing them.

To delete a universal share

1. Unmount the universal share on all the clients.
2. On the left, click **Storage > Disk storage**. Then click on the **Universal shares** tab.
3. Select the universal share that you want to delete and click **Delete > Delete**.

Mounting a universal share

Choose the mounting procedure that matches the type of universal share you created.

Mount a CIFS/SMB universal share

To mount an SMB universal share using Windows Explorer

1. Log on to the Windows server, then navigate to the **Map a Network Drive** tool.
2. Choose an available drive letter.
3. Specify the mount path as follows:

   `\\<MSDP storage server>\<id>`

   For example, `\\server.example.com\my-db-share`

   You can find the mount path on the NetBackup web UI: **Storage > Disk storage > Universal shares**.
4. Click **Finish**.

To mount an SMB universal share using Windows command prompt

1. Log on to the Windows server, then open a command prompt.
2. Specify the mount path using the following command:

   `net use <drive_letter>:\\<MSDP storage server >\<id>`

   For example: `net use <drive_letter>:\\<MSDP storage server >\<id>`
3. Specify the mount path as follows:

   `\\<MSDP storage server>\<id>`

   For example, `\net use \\server.example.com\my-db-share`

   You can find the MSDP storage server name and the export path from the Universal share details page in the NetBackup web UI: **Storage > Disk storage > Universal shares**

Mount an NFS universal share

To mount an NFS universal share

1. Log on to the server as root.
2. Create a directory for the mount point using the following command:

   `mkdir /mnt/<your_ushare_mount_point_subfolder>`
3. Mount the universal share using the following one of the following commands:

- NFSv3:

  ```
  mount -t nfs <MSDP storage server>:<export path> -o
  rw,bg,hard,nointr,rsize=1048576,wsize=1048576,tcp,actimeo=0,vers=3,timeo=600
  /mnt/<your_ushare_mount_point_subfolder>
  ```

  For example:

  ```
  mount -t nfs
  server.example.com:/mnt/vpfs_shares/3cc7/3cc77559-64f8-4ceb-be90-3e242b89f5e9
  -o
  rw,bg,hard,nointr,rsize=1048576,wsize=1048576,tcp,actimeo=0,vers=3,timeo=600
  /mnt/<your_ushare_mount_point_subfolder>
  ```

- NFSv4:

  ```
  mount -t nfs <MSDP storage server> : <export path> -o
  vers=4.0,rw,bg,hard,nointr,rsize=1048576,wsize=1048576,tcp,actimeo=0,
  vers=4,timeo=600 /mnt/ <your_ushare_mount_point_subfolder>
  ```

  Note: If you use NFSv4 on a Flex Appliance application instance, the export path must be entered as a relative path. Do not include `/mnt/vpfs_shares`.

  For example:

  ```
  mount -t nfs
  server.example.com:/3cc7/3cc77559-64f8-4ceb-be90-3e242b89f5e9
  -o
  rw,bg,hard,nointr,rsize=1048576,wsize=1048576,tcp,actimeo=0,vers=4,timeo=600
  /mnt/<your_ushare_mount_point_subfolder>
  ```

  For NetBackup FlexScale and AKS/EKS cloud platforms, if you use NFSv4 to mount the NFS share on NFS client, you must use the relative share path without the prefix `/mnt/vpfs_shares`.

  For example, if the export share path is engine1.com:/mnt/vpfs_shares/usha/ushare1, use NFSv4 to mount it on client as follows:

  ```
  mount -t nfs -o 'vers=4' engine1.com:/usha/ushare1
  /tmp/testdir.
  ```

- To mount an NFS share with Kerberos based authentication for NFSv3 or NFSv4, use any of the three types of security options:
  - krb5
  - krb5i
  - krb5p

NFSv3 with krb5:

```
mount -t nfs <MSDP storage server>:<export path> -o
rw,bg,hard,nointr,rsize=1048576,wsize=1048576,tcp,actimeo=0,vers=3,timeo=600,sec=krb5
/mnt/<your_ushare_mount_point_subfolder>
```

NFSv4 with krb5:

```
mount -t nfs <MSDP storage server>:<export path> -o
rw,bg,hard,nointr,rsize=1048576,wsize=1048576,tcp,actimeo=0,vers=4,timeo=600,sec=krb5
/mnt/<your_ushare_mount_point_subfolder>
```

- Use krb5 for authentication only.
- krb5i computes a hash on every remote procedure (RPC) call request to the server and every response to the client. The hash is computed on an entire message: RPC header, plus NFS arguments or results.
- krb5p uses encryption to provide privacy. With krb5p, NFS arguments and results are encrypted.

krb5 provides better performance and it decreases in the following order: krb5 > krb5i > krb5p.

You can find the mount path on the NetBackup web UI: **Storage > Disk storage > Universal shares**.

Creating a protection point for a universal share

You can create a protection point for the data in a universal share that lets you manage and protect the data in the share. Creating a protection point is accomplished by creating a Universal-Share backup policy.

If an MSDP storage server is configured with multiple universal shares, a single policy can be created for some or all of the shares. You can also create individual policies, one for each share. If multiple storage servers are configured with universal shares, each storage server should be configured with its own specific policy to protect the universal shares on that storage server.

See “Overview of universal shares” on page 601.

Protection point - cataloging and protecting universal share data

Any data that is initially ingested into a universal share resides in the MSDP located on the appliance-based media server that hosts the universal share. This data is not referenced in the NetBackup Catalog and no retention enforcement is enabled. Therefore, the data that resides in the universal share is not searchable and cannot be restored using NetBackup. Control of the data in the share is managed only by the host where that share is mounted.

The protection point feature supports direct integration with NetBackup. A protection point is a point-in-time copy of the data that exists in a universal share. Creation and management of a protection point is accomplished through a NetBackup policy, which defines all scheduling and retention of the protection point. The protection point uses the Universal-Share policy, which can be configured through NetBackup web UI. After a protection point for the data in the universal share is created, that point-in-time copy of the universal share data can be managed like any other protected data in NetBackup. Protection point data can be replicated to other NetBackup Domains or migrated to other storage types like tape or cloud, using storage lifecycle policies. Each protection point copy is referenced to the name of the associated universal share.

To create a protection point policy for a universal share

1. Create a policy using the NetBackup web UI.
2. On the **Attributes** tab, select **Universal-Share** from the **Policy type** list.

   For the **Policy storage**, you must use the storage unit that hosts the universal share. You must create one if one does not exist.

   If multiple storage servers are configured with universal shares, each of the storage servers should be configured with its own specific policy. This configuration ensures that the universal shares on that storage server are protected.
3. Under **Destination**, select storage unit from the **Policy storage** list.

   See *Policy storage (policy attribute)* in *NetBackup Administrator's Guide Volume I* for more information about policy storage setting.

   The storage unit for universal share policy must be in the same disk pool volume where the universal share is created.
4. On the **Schedules** tab, select either **FULL** or **INCR**.
5. On the **Clients** tab, enter the name of the desired client.

   Universal share is an agentless technology, so the client name that is specified is used only for cataloging purposes. You can enter a NetBackup Appliance, NetBackup Virtual Appliance, Flex Appliance media server application instance, or MSDP BYO server name or a host where universal share is mounted. The client name can be a short name, Fully Qualified Domain Name (FQDN), or IP address.

6 On the **Backup Selections** tab, enter the path of the universal share.

You can find the export path from the Universal share details page NetBackup web UI: **Storage > Storage Configuration > Universal Share**.

For example:
`/mnt/vpfs_shares/3cc7/3cc77559-64f8-4ceb-be90-3e242b89f5e9`

You can use the `NEW_STREAM` directive if you require multistream backups.

You can also use the `BACKUP X USING Y` directive, which allows cataloging under a different directory than the universal share path. For example: `BACKUP /database1 USING /mnt/vpfs_shares/3cc7/3cc77559-64f8-4ceb-be90-3e242b89f5e9`. In this example, the backup is cataloged under `/demo/database1`.

7 Run the **Universal-Share** policy.

After the backups are created, you can manage the backups with NetBackup features, such as restore, duplication, Auto Image Replication, and others.

You can instantly access backup copies from local LSU or cloud LSU with web UI or NetBackup Instant Access APIs.

For more information about instant access for cloud LSU:

See "About instant access for object storage " on page 410.

For information about NetBackup APIs, see the following website:

https://sort.veritas.com/netbackup/documents

Select NetBackup and then the version at the bottom of the page.

Restoring data using universal shares

You can restore data from a universal share backup.

Protection point restores

Restoring data from a protection point is exactly the same as restoring data from a standard client backup. The standard Backup, Archive, and Restore interface or the NetBackup web UI can be used to restore data. The client name that is used for the restore is the universal share name in the Universal-Share policy. Alternate client restores are fully supported. However, to restore to the system where the universal share was originally mounted, NetBackup client software must be installed on that system. This software is necessary since a NetBackup client is not required to initially place data into the universal share.

NetBackup also supports a wide variety of APIs, including an API that can be used to provision (instant access) or create an NFS/SMB share that is based on any protection point point-in-time copy. This point-in-time copy can be mounted on the originating system where the universal share was previously mounted. It can be provisioned on any other system that supports the mounting of network share. NetBackup client software is not required on the system where the provisioned share is mounted.

Restore methods of universal shares

Besides offering a fast data protection process, the Protection Point offers two powerful restore methods:

Client-based restore:

- Data protected using a Protection Point is restored using the exact same method as restoring data from a standard client backup:
  - Restore to the original universal share.
    In this case, the original universal share must be present. Specify the universal share path as the restore destination and the media server where universal share resides as the client. However, for large data restores, consider restoring to an alternate location.
  - Restore to an alternate location.
    A standard NetBackup client must be installed on any system where the restore is directed.

Provisioned restore (Instant Access):

- A Protection Point is a point-in-time (PIT) copy of the data as it existed on the Universal Share when any Protection Point was initiated. This PIT copy of the data can be exported as a separate network share of the Protection Point data. This PIT copy of the Projection Point is called a provisioned copy of the data. The data in this provisioned share is not necessarily connected to any data in the primary universal share. It can be used as an autonomous version of the PIT Protection Point data. Any changes to this provisioned copy of the data have no effect on data in the original universal share. It also does not have any effect on the source PIT copy of the data.
  The PIT copy can be mounted on the originating system where the universal share was previously mounted. It can also be provisioned on any other system that supports the mounting of a network share. In this sense, the NetBackup Protection Point provides a method of copy data management that offers you another powerful way of using the data that is managed with NetBackup. The process of provisioning a Protection Point is performed using a NetBackup API.

This API and all NetBackup APIs are described in the *NetBackup API Reference* documentation, which is located on the NetBackup primary server.

Disaster recovery for a universal share

Universal share disaster recovery is available for BYO, AKS, and EKS environments when data in a share or share-specific metadata is corrupted or deleted.

Before beginning this procedure, verify that data was backed up with a cloud-configured universal share and at least one PIT image exists for each share that is to be recovered. If there is no PIT image, this procedure cannot be used. The host name of the computer the disaster recovery is performed on must match the host name the shares were originally created on.

If the following procedure is performed after a regular MSDP disaster recovery, ensure that NGINX, SPWS, NFS, and SMB are configured as described previously in this chapter.

Performing disaster recovery for all universal shares

1. Ensure that the `vpfs_cloud.cfg` file exists at the following location:

   `[MSDP storage directory]/etc/puredisk/vpfs_cloud.cfg`

   If the file does not exist, run the following command:

   `/usr/openv/pdde/vpfs/bin/vpfscld --download_vpfs_cloud_cfg`

2. Ensure that the nfs export list exists at the following location:

   `[MSDP storage directory]/etc/vpfs-shares.exports` on BYO or `[MSDP storage directory]/cat/config/vpfs-shares.exports` on AKS or EKS if there are any loud-configured NFS shares.

   If the list does not exist, run the following command:

   ```
   /usr/openv/pdde/vpfs/bin/vpfscld --download_export_list
   --share_type nfs
   ```

3. Ensure that the smb export list exists at the following location:

   `[MSDP storage directory]\/etc/vpfs-shares.conf` on BYO or `[MSDP storage directory]/cat/config/samba/vpfs-shares.conf` on AKS or EKS if there are any cloud-configured SMB shares.

   If the list does not exist, run the following command:

   ```
   /usr/openv/pdde/vpfs/bin/vpfscld --download_export_list
   --share_type smb
   ```

4 Restore data to the shares and recover the metadata with the following command:

```
/usr/openv/pdde/vpfs/bin/vpfs_actions -a disasterReovery
--cloudVolume CLOUDVOLUMENAME
```

Where `CLOUDVOLUMENAME` is the name of the MSDP cloud volume containing the shares you want to restore. If you want to restore shares across multiple cloud volumes, run this command for each cloud volume.

5 Restart the NetBackup services using the following commands:

```
/usr/openv/netbackup/bin/bp.kill_all
```

```
/usr/openv/netbackup/bin/bp.start_all
```

Disaster recovery for universal shares on migrated cloud LSU

The following procedure should be performed after the regular MSDP cloud LSU consolidation.

These universal share disaster recovery steps are supported in BYO, AKS, and EKS environments. For BYO deployments, ensure all prerequisites are met before proceeding with the steps below. See "Prerequisites to configure universal shares" on page 604.

The data from universal shares can only be recovered if at least one point-in-time (PIT) image exists for each share. If no PIT image is available, only the configurations are recoverable. No actual data will be present on the recovered shares.

To verify if the PIT image exists, run the following command:`/usr/openv/pdde/vpfs/bin/vpfscld --list`

Note: The migration steps do not retain the original share quota settings. You must manually reset the quotas to their original values after a successful migration.

To recover universal shares on migrated cloud LSU

1. Recover universal share configurations.

   ```
   /usr/openv/pdde/vpfs/bin/vpfscld --migrated_cloud_lsu --lsu <lsu name> --old_sts <old storage server name>
   ```

   Ensure that there are no duplicate share names between the source and target storage servers to avoid errors. If duplicates exist, back up shares on both servers and delete duplicates from either the source or target.

2. Recover NFS share configurations if previously configured on the old storage server:

   ```
   /usr/openv/pdde/vpfs/bin/vpfscld --migrated_cloud_lsu --download_export_list --share_type nfs --lsu <lsu name> --old_sts <old storage server name>
   ```

3. Recover SMB share configurations if previously configured on the old storage server:

   ```
   /usr/openv/pdde/vpfs/bin/vpfscld --migrated_cloud_lsu --download_export_list --share_type smb --lsu <lsu name> --old_sts <old storage server name>
   ```

   Ensure that SMB users are configured on the target storage server, and their credentials match those on the source storage server.

   For more information on authentication setup, See "Configuring universal share user authentication" on page 607.

4. Restore data to the shares and recover the metadata:

   ```
   /usr/openv/pdde/vpfs/bin/vpfs_actions -a disasterRecovery --cloudVolume CLOUDVOLUMENAME --migratedCloudLsu --oldSts <oldSts>
   ```

   - `CLOUDVOLUMENAME`: Name of the MSDP cloud volume containing the shares you want to restore. If restoring shares from multiple cloud volumes, run this command separately for each volume.
   - `oldSts`: The name of the old storage server from which the cloud LSU was migrated.

Note: (Cloud Scale only) Run the following command to ensure that OCSD process can download and write configuration files to the target MSDP Kubernetes pod disk volume. This is necessary because MSDP processes run as a non-root user:

```
chown -R msdpsvc:pduser /msdp/data/dp1/pdvol/meta_dir/
```

After completing the disaster recovery steps, clients may encounter "permission denied" errors when accessing the content of SMB universal shares. One possible reason is a mismatch in the UID/GUID of the owner user between the source and target storage servers. To resolve this, ensure that ownership and permissions of the universal share content are correctly set on the target storage server.

For example, On the MSDP server, verify that the universal share smb1-test has the correct owner (smb_user) and appropriate permissions.

- Change the permission: `chmod -R 755 /mnt/vpfs_shares/Ia-s/smb1-test`
- Change the owner: `chown -R smb_user:smb_user /mnt/vpfs_shares/Ia-s/smb1-test`

5 Restart the NetBackup services:

```
/usr/openv/netbackup/bin/bp.kill_all

/usr/openv/netbackup/bin/bp.start_all
```

6 Clean up the existing universal shares that are associated with the migrated cloud LSU on the old storage server.

7 On the client, unmount the old mount points and re-mount using the new export paths.

Restore universal share data using Instant Access or single file recovery

This feature allows a workload administrator to restore universal share data using Instant Access or single file recovery in the NetBackup web UI. Instant Access recovery would enable the creation of a universal share to mount the data and present a path to the end user to access files. A workload administrator that does not have administration rights can restore the data without intervention from a NetBackup administrator.

When replication is performed, a universal share asset is created on the target server that is based on the information in the image. NetBackup automatically deletes a universal share if there are no recovery points and no corresponding universal share storage.

Performing a universal share self-service recovery

This procedure allows a workload administrator with the required RBAC permissions/roles to choose one asset and select recovery points and all available

copies of that recovery point. They can also select a default copy or copy of interest to perform provisioning of a universal share.

Recover a universal share

1. In the NetBackup web UI, select **Workloads > Universal shares**.
2. In the **Shares** tab, select the universal share you want to use.
3. Select **Recovery points** to view all recovery images for that universal share.
4. Click **Recover** and then click on **Create instant access universal share** on the image you want to use.
5. Provide the following required information:
   - Enter a **Display name**. This name is used in the universal share path.
   - Select the **Protocol**: NFS or SMB (CIFS)
   - Specify a **Host** that is allowed to mount the share and then click **Add to list**. You can use the host name, IP address, short name, or the FQDN to specify the host. You can enter multiple hosts for each share.
6. Click **Save**.
7. NetBackup creates a recovery job. You can view this job by clicking on **Restore activity**.
8. Select **Workloads > Universal shares > Instant access universal shares** to review the universal share.

Advanced features of universal shares

You can leverage the following advanced features of universal shares:

- Universal share with object store
- Universal share accelerator
- Ingest mode
- Direct Network File System (dNFS)
- MSDP data volumes
- Universal share WORM capability
- Universal share scale out

Direct universal share data to object store

Universal Shares with object store provides the ability for data in the universal shares to be directed to object storage in a deduplicated format.

Table 15-7 Supported platforms for universal shares with object store

Supported platforms	Description
Azure Kubernetes Service (AKS)	This platform is supported and enabled by default.
Amazon Elastic Kubernetes Service (EKS)	This platform is supported and enabled by default.
VM in Azure or AWS	This platform is supported. You must manually enable this option. See "Enabling a universal share with object store" on page 640.
RHEL 7.6+/8/9 on premise (Object storage on premise)	This platform is supported. You must manually enable this option.

Lifecycle management of a universal share with object store

The three phases of the lifecycle management:

- Create a universal share: Refer to the section *Create a universal share* in the NetBackup Web UI Administrator's Guide for more information about how to create a universal share.
- Delete a universal share: If you delete the universal share in the NetBackup web UI, the MSDP cloud volume cannot be deleted if it is in use by any universal shares or instant access. If you delete a universal share, be aware that all of the snapshots from the universal share are expired on the local disk and the cloud bucket. You must log in to the media server to expire all the snapshot copies from the universal share before you delete it.
- Snapshot retention and lifecycle: The `vpfsd` command runs a background thread and continues to monitor the retention of the snapshot copies. The `vpfsd` command expires the snapshot when the retention time is reached. The retention for the full snapshot and incremental snapshot can be configured in the `vpfsd_config.json` file.

Enabling a universal share with object store

A universal share or instant access with object stores are enabled by default on AKS or EKS. But for the cloud virtual machines to enable the object store feature, you must manually enable this feature.

Enable a universal share with object store

1. Add the `universal-share-object-store = 1` option into the `etc/msdp-release` file.

   Example:

   ```
   cat /etc/msdp-release
   universal-share-object-store = 1
   ```

2. Verify that the `UNIVERSAL_SHARE_OBJECT_STORE` name is in the `extendedcapabilities` option.

   Example:

   ```
   nbdevconfig -getconfig -stype PureDisk
   -storage_server your_storage_server_name
   |grep UNIVERSAL_SHARE_OBJECT_STORE
   ```

3. On the media server or the primary server, run the following commands to reload the storage server attributes:

   ```
   nbdevconfig -getconfig -stype PureDisk
   -storage_server your_storage_server_name > /tmp/flags

   nbdevconfig -setconfig -stype PureDisk
   -storage_server your_storage_server_name -configlist /tmp/flags
   ```

The following are optional parameters you can add to the universal share with object store. These options are located in:
storage_path`/etc/puredisk/vpfsd_config.json`

Snapshot retention:

- `"cloudFullTaskInterval": 36000,`: Automatically creates the full snapshot for the universal share interval and the default value is 10 hours. This entry must be an integer using the unit of seconds.
- `"cloudIncrTaskInterval": 1800,`: Automatically creates the incremental snapshot for the universal share interval and the default value is 30 minutes. This entry must be an integer using the unit of seconds.

- `"cloudFullSnapshotRetention": 172800,`: The retention time of the full snapshot copy. When the retention expires, the full snapshot is deleted from local storage and the cloud bucket storage. The default value is 48 hours. If the retention is set longer than 48 hours, there might be an effect on space reclamation.

Local disk cache configuration:

- `"CloudCacheSize": 500,`: The local disk cache size for the universal share and instant access. This option applies only to the universal share with object store and instant access with object store. The `vpfsd` command removes this amount of space from the `spoold` service so you must verify there is enough space for the cache size. Otherwise, the universal share with object store or instant access for the object store are not created. MSDP verifies that there is enough configured space before the creation of the universal share. You must restart `vpfsd` when you increase the size of the cache and be aware that if there is not enough free space for the cache size, `vpfsd` cannot start.

  After the universal share or instant access with object store is removed, that amount of space does not return to `spoold` automatically. Reduce the `CloudCacheSize` to return some space back to `spoold`. You must restart `vpfsd` after the removal.

- `"CloudCacheLowThreshold": 50,`: The `vpfsd` service starts to reclaim the space of the cloud cache when the space usage of the cache reaches the low threshold. This entry is in the unit of percentage.

- `"CloudCacheHighThreshold": 85,`: The `vpfsd` service stops any data from being written or the download of any data when the space usage of the cache reaches the high threshold. The data write and the data download continue when there is some free space. This entry is in the unit of percentage.

Local disk cache sizing:

- The size of the cache depends on number of the data volumes, number of the universal shares, and the total number of the files in all of the universal shares. There are two types of data stores in the local disk cache: the metadata and the deduplicated segments in the data containers. A greater number of files requires more disk cache, and a greater number of data volumes also requires more disk cache.

  Each one million files requires about 300 GB disk cache for metadata of universal share. For the data container cache, each universal share has 32 data container caches for each data volume by default, and there are 1024 data containers for the read cache by default for all the universal shares.

  The following is the algorithm to calculate the disk cache size:

  Metadata size: 300 GB * Number of File (unit is million)

Data container size: 64 MB * (32 * Number of data volume * number of share + 1024)

Total cache size: (metadata size + data container size) / 0.85 (`CloudCacheHighThreshold`)

The default size of the local disk is 500 GB. This size can handle most scenarios. The configuration below is the recommend setting if the number of data volumes or partition is 4 or less. Configure `CloudCacheSize` in `vpfsd_config.json` to tune the cache size.

Number of files (in millions)	Number of shares	Number of partitions	Disk cache size (in GBs)
0.5	1	4	400
0.5	20	4	400
1	1	4	600
1	20	4	600
2	1	4	800
3	1	4	1200
4	1	4	1600
5	1	4	2000
5	20	4	2000
6	1	4	2200
7	1	4	2600
8	1	4	3000
9	1	4	3400
10	1	4	3800
10	20	4	3800

Snapshot management:

- List all of the snapshots which include the full snapshot and incremental snapshot in the cloud bucket:

```
/usr/openv/pdde/vpfs/bin/vpfscld -list
```

- Manually take snapshot and upload snapshot and data to cloud bucket:

  ```
  /usr/openv/pdde/vpfs/bin/vpfscld --snapshot
  --share_id <share> --snap_type <full|incr>
  ```

- Manually remove a snapshot from local and cloud, please be aware of that the expired snapshot is not recoverable:

  ```
  /usr/openv/pdde/vpfs/bin/vpfscld --expire
  --share_id <share> --pit <point in time>
  ```

- Manually recover a snapshot from a cloud bucket:

  ```
  /usr/openv/pdde/vpfs/bin/vpfscld -recover
  --share_id <share> [--tgt_id <target>] [--pit <point in time>]
  [--force]
  ```

- Configure the number of snapshot copies for universal share.
  The default number of snapshot copies is 2. It can be configured with the `cloudMaxSnapshotCopy` parameter in the `vpfsd_config.json`. When the value of `cloudMaxSnapshotCopy` set to 0, it disables the feature that automatically takes the snapshot. Higher values require more local disk cache if the universal share stores the snapshot on local disk cache.
- Configure universal share stores snapshot copies on the local disk cache.
  With NetBackup 10.4 and later versions, snapshots will be stored only on the cloud bucket instead of being stored on the local disk cache. However, you can configure and control the storage of additional snapshots on the local disk cache. If you still want to keep the snapshot on the local disk cache, you must configure `cloudKeepLocalSnapshotCopy` as `true` in `vpfsd_config.json`. It also requires more local disk space.

Note: To enable object store for universal share and instant access, add `universal-share-object-store = 1` and `instance-access-object-store = 1` to `/etc/msdp-release`.

Enabling instant access with object storage

Enable instant access with object storage

1. Add `instant-access-object-store = 1` into the `/etc/msdp-release` file.

   Example:

   ```
   /etc/msdp-release
    instant-access-object-store = 1
   ```

2. Verify that the capability `IA_OBJECT_STORE` is in the `extendedcapabilities`.

   ```
   nbdevconfig -getconfig -stype PureDisk -storage_server
   <your_storage_server_name> |grep IA_OBJECT_STORE
   ```

3. On the media server and the primary server, run the following commands to reload the storage server attributes:

   ```
   nbdevconfig -getconfig -stype PureDisk
   -storage_server <your_storage_server_name> > /tmp/flags
   nbdevconfig -setconfig -stype PureDisk
   -storage_server <your_storage_server_name> -configlist /tmp/flags
   ```

4. Add `universal-share-object-store = 1` and `instant-access-object-store = 1` to `/etc/msdp-release` to enable object store for universal share and instant access.

Universal share accelerator for data deduplication

The universal share accelerator feature directly provides the file system mount point on the workload or client. It does not provide the NFS/SMB service. The data does not go to the storage server. The data goes directly to the cloud bucket and only sends the fingerprint-related metadata to the storage server to do the data deduplication. The data directly comes from cloud bucket when read data from an accelerator.

The universal share accelerator leverages the universal share with object store feature. The major difference is that universal share with object store is on the storage server side and universal share accelerator is on the client side. The universal share accelerator is delivered with NetBackup client packages. It requires you to install the NetBackup client on the respective workload with universal share accelerator feature enabled before user can create a universal share accelerator from the NetBackup web UI. Once the universal share accelerator has been created on the NetBackup web UI then you can configure the accelerator on the client side.

Supported platforms

- Client:
  - RHEL 7.6 and newer, RHEL 8.x and RHEL 9.x
    - Only supported in AWS or Azure.
- Storage server:
  - Storage server version 19.0 and later
  - RHEL 7.6 and newer, RHEL 8.x and RHEL 9.x

    Note: Universal share accelerator is not supported on SUSE Linux Enterprise.
  - AKS/EKS or MSDP cluster with AWS/Azure
- NetBackup:
  - Primary server: 10.3 and newer
  - Media server: 10.3 and newer
  - Client: 10.3 and newer

Limitations

- Universal share accelerator doesn't support multiple `vpfsd` instances.
- DR is not supported for universal share accelerator.

Preparing NetBackup for the universal share accelerator

Retrieve the AWS, Azure S3, or blob account and keys to create the MSDP Cloud LSU.

The disk size for the universal share accelerator local disk cache (**<storage-path>**) should be at least 500GB. The amount of disk is used as the local disk cache of universal share accelerator, it should be a good performance disk. It can be a mount point with separate disk or one folder of existing mount point. If the local disk cache or storage is shared with other applications or systems, NetBackup ensures that it can use the configured amount of disk space. The cloud cache size (`CloudCacheSize`) can be found in *<storage-path>*`/etc/puredisk/vpfsd_config.json`.

Cohesity recommends the use of separate mount points for the universal share accelerator.

Ensure the following:

- You have sufficient, free usable space for the universal share accelerator if it uses a shared disk or mount point with other applications.
- You use a Fully Qualified Domain Name (FQDN) as the client hostname.

Installing the universal share accelerator

The install process is very similar with NetBackup client installation, the details about the NetBackup client installation can be found at the following links:

- About NetBackup client installation on UNIX and Linux
- Installing UNIX clients locally

To install the universal share accelerator

1. Add the option `INCLUDE_VRTSPDDEU_CLIENT` into `NBInstallAnswer.conf` file.

   To automatically install the binaries for accelerator you need to add the option `'INCLUDE_VRTSPDDEU_CLIENT = INCLUDE'` into `/tmp/NBInstallAnswer.conf`. The binary does not install it by default.

2. Add the `CLIENT` entry with the workload `FQDN` into the NetBackup primary server `/usr/openv/netbackup/bp.conf` file.

3. Copy the NetBackup Linux client package to the workload server.

   About NetBackup client installation on UNIX and Linux

   For more information about how to install the client on UNIX and Linux, refer to the `NetBackup Installation Guide`.

Creating a protection policy for the universal share accelerator

To create a protection policy for the universal share accelerator

1. Create a policy using the NetBackup web UI.
2. On the **Attributes** tab, select **Universal-Share** from the **Policy type** list.

   For the **Policy storage**, you must use the storage unit with the same disk volume with universal share accelerator. You must create one if it doesn't exist.

3. Under **Destination**, select storage unit from the **Policy storage** list.

   See *Policy storage (policy attribute)* in *NetBackup Administrator's Guide Volume I* for more information about policy storage setting.

   The storage unit for universal share policy must be in the same disk pool volume where the universal share is created.

4 On the **Schedules** tab, select either **Full backup** or **Differential incremental backup**.

5 On the **Clients** tab, click **Add** and enter the host name of the client in **Client name**.

The host name requires the same host name of NetBackup client, you can find the host name on NetBackup web UI **Hosts > Host properties**. Or open the universal share accelerator details page and find the **host name** in the section **Hosts**.

6 On the **Backup Selections** tab, click **Add** and enter the path of the universal share in **Pathname or directive** and then click **Add to list**.

You can find the export path from the universal share details page NetBackup web UI: **Storage > Disk storage > Universal shares**. For example: `/mnt/vpfs_shares/accl/accl`

You can use the **NEW_STREAM** directive if you require multistream backups.

You can also use the **BACKUP X USING Y** directive, which allows cataloging under a different directory than the universal share path. For example: `BACKUP /demo/database1 USING /mnt/vpfs_shares/accl/accl`. In this example, the backup is cataloged under `/demo/database1`.

7 Run the **Universal-Share** policy.

After the backups are created, you can manage the backups with NetBackup features, such as restore, duplication, Auto Image Replication, and others.

Configure a universal share accelerator

To create a universal share accelerator on a build your own (BYO), it requires you to enable the feature universal with object store, refer to the following topic to enable it. The universal share accelerator also requires a cloud disk volume or a cloud disk pool.

See “Enabling a universal share with object store” on page 640.

Creating a universal share accelerator

Log on to the NetBackup web UI and create a universal share. Reference the *Create a universal share* section in the *NetBackup Web UI Cloud Administrator’s Guide* for more information.

When you create a universal share accelerator, input the following information:

- **Display name**: The share ID which is used to mount on the NetBackup client or workload.
- **Type**: Choose **Accelerator**.

- **Storage server**: The storage server host name.
- **Disk volume**: The MSDP cloud disk volume.
- **Quota**: Unlimited by default.
- **Hosts**: The NetBackup client host name.

Mounting a Universal share accelerator

First, configure the universal share accelerator using the `vpfs_accelerator.sh` script on the NetBackup client computer using the following:

```
/usr/openv/pdde/vpfs/bin/vpfs_accelerator.sh --config
--storage-server=<storage server> --engine-host=<hostname> --mode=byo
--storage-path=<path>
```

- `storage-server`: The storage server name.
- `engine-host`: Engine name of the cluster or the storage server of the standalone server.
- `mode`: Accelerator mode, it should be **byo**.
- `storage-path`: List of local storage paths, comma separated with no spaces, and those paths should be pre-created.

Next, mount the Universal share accelerator using the following:

```
/usr/openv/pdde/vpfs/bin/vpfs_accelerator.sh --create --share-id=<ID>
--cloud-volume=<MSDP cloud disk volume>
```

- `share-id`: The universal share ID that can be found in the section storage configuration and the universal shares tab in the NetBackup web UI.
- `cloud-volume`: The cloud volume that is the column **Volume** in the universal shares tab.

Deleting a universal share accelerator

To delete a universal share accelerator, use the following:

1. Stop the universal share accelerator on the client and delete it.
   - `/usr/openv/pdde/vpfs/bin/vpfs_accelerator.sh --stop --share-id=<id>`
   - `/usr/openv/pdde/vpfs/bin/vpfs_accelerator.sh --delete --share-id=<id>`
   - `/usr/openv/pdde/vpfs/bin/vpfs_accelerator.sh --stop-all`

2. In the NetBackup web UI, open the universal share list page and select the accelerator and delete it.

Unconfiguring a universal share accelerator

When you unconfigure the universal share accelerator on the client side, you delete the universal share accelerator and its point in times. Unconfiguring removes the connection with the NetBackup primary server but the backup images on the primary server side are not deleted. To delete the universal share accelerator, you must delete it from the NetBackup web UI.

To unconfigure the universal share accelerator on the client side, use the following:`/usr/openv/pdde/vpfs/bin/vpfs_accelerator.sh –unconfig`

Note: If you run `vpfs_accelerator.sh –unconfig` it deletes the data for the universal share accelerator and the data cannot be recovered.

Managing the universal share accelerator services

Use the following to manage the universal share accelerator services:

- `vpfs_accelerator.sh --start --share-id=<id>`: Start the universal share accelerator.
- `vpfs_accelerator.sh --stop --share-id=<id>`: Stop the universal share accelerator.
- `vpfs_accelerator.sh --start-all`: Start all services.
- `vpfs_accelerator.sh --stop-all`: Stop all services.

Adding additional storage paths for universal share accelerator

Use the following procedure to add a new partition.

To add additional storage paths for universal share accelerator

1. Shut down the universal share accelerator.

   ```
   vpfs_accelerator.sh --stop-all
   ```

2. Edit the `edit_fstab` to add new partition.

   ```
   /usr/openv/pdde/vpfs/bin/edit_fstab [partition_1] [partition_2]
   ... [partition_N] Where partition_1 - partition_N are the
   partition paths to be added to the fstab.cfg file. Partitions
   that already exist in fstab.cfg are ignored.
   ```

   Example: Add new partition `/msdp1/`.

   ```
   edit_fstab /msdp1/
   ```

3. Start up the universal share accelerator.

   ```
   vpfs_accelerator.sh --start-all
   ```

About the universal share accelerator quota

The quota can be enabled in the NetBackup web UI, you can increase or decrease the quota. The NetBackup web UI doesn't support disabling the quota. When you remove a sparse file, the quota may not reflect the removed amount space which the sparse file uses.

Enabling or changing the quota

To enable or change quota

1. Open the NetBackup web UI.
2. On the left, click **Storage > Disk storage**.
3. On the **Universal shares** tab, click the **Display name** to view the details of the universal share.
4. Find the section **Quota** and click **Edit**.
5. Input the quota value and select the unit GB or TB. The minimum size of the quota is 1 GB, and the maximum size should not be larger than the usable size of the storage system.

Reviewing the quota usage

Use one of the following to review the quota usage:

1. Check the quota usage in the NetBackup web UI, on the **Disk storage** page in the **Universal shares** tab. The column **Quota used** shows the current quota usage.

2. Run the `df` command with the mount point or drive property if it's an SMB share mounted on windows.

   Example:

   ```
   # df -h /mnt/vpfs_shares/test/test
   Filesystem Size  Used Avail Use% Mounted on
   vpfsd      2.0T  1.9G  2.0T   1% /mnt/vpfs_shares/test/test
   ```

3. Use the command `vpfs_quota` to query the quota usage. The command should be run on the workload computer if it's a universal share accelerator. The output comes from the command `vpfs_quota` and may not be the same as in the web UI.

   ```
   vpfs_quota
   Usage:
   vpfs_quota <status> <share_id>
   ```

   Example:

   ```
   /usr/openv/pdde/vpfs/bin/vpfs_quota status test
               Quota: 2199023255552
                Used: 1933574144
                Enabled: Yes
   ```

Repairing the quota of the universal share

In some situations, the quota file maybe corrupted or the usage quota is not correct. You can use `/usr/openv/pdde/vpfs/bin/vpfs_quota repair <share ID>` to repair the quota for the universal share. Use the following procedure to repair the quota for a share.

To repair the quota of the universal share

1 Ensure there is no write operations on the universal share.

2 Make one backup copy of the `quota.dat` for `<storage>/meta_dir/<share_dir>/<share_id>/quota.dat`.

Example:

```
mv /msdp/meta_dir/test/test/quota.dat
/msdp/meta_dir/test/test/quota.dat.bak
```

3 Use the command `vpfs_quota repair` to repair the quota.

Example:

```
/usr/openv/pdde/vpfs/bin/vpfs_quota repair test
```

4 Restart the NetBackup service.

- NetBackup appliance, BYO, or Flex media: `netbackup stop/start`
- Flex WORM, Flex scale, AKS, or EKS: `dedupe vpfs stop, dedupe vpfs start`

Note: This procedure doesn't support the quota repair on the deduplicate shell, to repair the quota on WORM storage server requires you to unlock the appliance.

Recovering a point in time for the universal share accelerator

To recover a point in time or snapshot for the universal share accelerator it requires some work on the storage server side to grant the share ID permissions. These permissions are so the client-side accelerator can use that share ID to connect to the storage server.

Create an accelerator from a point in time or snapshot

Use the following steps to recover one snapshot and start it with universal share accelerator. The ID of `usa_snap` is used as the new accelerator ID in the following procedures.

On the storage side

- Register the new share ID, which is share ID for the accelerator.

```
vpfscld --manage_accl_id --reg --share_id usa_snap --mode byo
--lsu labvol --client usademo.name.name.domain.com
```

On the client side:

1 Recover the snapshot from the cloud bucket. Use the `vpfscld --list` command to find the snapshot PIT before recovery.

```
vpfscld --list
/msdp/meta_dir/usa/usa
Name: pit_0f0430a8-4cea-41dc-8e22-ed1b6f7804e9
Type: full
Create_time: 1683642234
```

- `Name`: The snapshot ID, which can be used to recover the share.

- `Type`: The snapshot type, it can be full and incremental.
- `Create_time`: The snapshot creation time.

2 Recover on the client.

```
vpfscld --recover
--share_id usa
--tgt_id usa_snap
--pit pit_0f0430a8-4cea-41dc-8e22-ed1b6f7804e9
--lsu labvol
```

- `share_id`: The original universal share accelerator share ID.
- `tgt_id`: The target share ID.
- `pit`: The snapshot ID which is used to do the recovery.
- `lsu`: The cloud volume name.

3 Start the new accelerator using the following:

```
vpfs_accelerator.sh --start --share-id usa_snap
```

4 The new mount point can access all the files in the accelerator.

Example:

```
/mnt/vpfs_shares/usa_/usa_snap
```

Deleting a recovered universal share accelerator

The NetBackup web UI does not manage the recovered universal share accelerator. It requires you to manually delete it. Use the following procedures to delete a recovered accelerator.

The ID of `usa_snap` is used as the new accelerator ID in the following procedures.

On the client side:

1 Stop the universal share accelerator.

```
vpfs_accelerator.sh --stop --share-id=usa_snap
```

2 Delete the accelerator.

```
vpfs_accelerator.sh --delete --share-id=usa_snap
```

On storage server side:

- Unregister the share ID.

```
vpfscld --manage_accl_id --unreg --share_id usa_snap --mode byo
--client usademo.name.name.domain.com
```

Logging for universal share accelerator

The log message or log files for universal share accelerator can be on the client, storage server, media server, and primary server. The following are the locations for the log files.

- Client:
  - **<storage-path>**/log/
  - /usr/openv/netbackup/logs/
- Storage/media server:
  - **<storage-path>**/log
  - /usr/openv/netbackup/logs/
  - /usr/openv/logs/
- Primary server:
  - /usr/openv/netbackup/logs/
  - /usr/openv/logs/

Ingest data to a universal share with the ingest mode

The purpose of the ingest mode of universal share is to dump data or to load backup data from a workload to the universal share over NFS/CIFS. When the ingest mode is turned on, a backup script prompts the universal share to persist all the data from memory to disk on the client side at the end of the backup or the dump.

The ingest mode differs a bit from the normal mode of a universal share. The ingest mode requires an additional operation to make sure the rest of the backup data or the dump data is persisted to the disk in the universal share. Every 60 seconds, a background job periodically flushes and persists the ingested data to disk.

The ingest mode is faster than normal mode as it does not guarantee all the ingested data is persisted to disk until the ingest mode is turn off. Therefore, turning ingest mode off is critical for data dump integrity.

Using the ingest mode

1. Create the universal and mount it on the client side. The protocol can be NFS or CIFS/SMB.
2. Turn on the ingest mode.

   You can turn on the ingest mode for a specific share on the NFS/SMB client side. In this case, the ingest mode applies only to the specified share.

For example, you can use the following commands to turn on the ingest mode on the Linux/Unix or windows:

- On Linux/Unix over NFS:

```
(echo [vpfs]&& echo ingest_mode=on) >
 <nfs_mount_point>/.vpfs_special_control_config
```

- On Windows over CIFS/SMB:

```
(echo [vpfs]&& echo ingest_mode=on) >
 <driver_letter>/.vpfs_special_control_config
```

3. Backup data or dump data to the universal share.
4. Turn off the ingest mode on the NFS/SMB client side, after the backup or dump is completed. For example:

- On Linux/Unix over NFS:

```
(echo [vpfs]&& echo ingest_mode=off) >
 <nfs_mount_point>/.vpfs_special_control_config
```

- On Windows over CIFS/SMB:

```
(echo [vpfs]&& echo ingest_mode=off) >
 <driver_letter>/.vpfs_special_control_config
```

Make sure to check the return value of the commands. If the return value is not 0, the data might have not been persisted successfully. In that case, you must back up or dump the data again.

Take universal share snapshot over NFS or SMB

This feature provides the ability to take a full or an incremental snapshot. When a full snapshot is taken, `vpfsd` makes the metadata and data consistent and then uploads the data to the cloud bucket. When an incremental snapshot is taken, `vpfsd` identifies the changed metadata and data since last full or incremental snapshot. Those changes are uploaded to the cloud bucket.

Taking a snapshot over NFS or SMB

- Take the snapshot from the client side.

  For the snapshot, add the following key-value pairs content into the `vpfs_special_control_config` control file:

  ```
  snapshot=[full/incr]
  ```

  Example:

  ```
  (echo [vpfs]&& echo snapshot=full && echo ingest_mode=off) >
  <nfs_mount_point>/.vpfs_special_control_config
  ```

Trigger universal share backup over NFS or SMB

This feature provides the ability to run a NetBackup backup policy for a Universal Share using `vpfs`. You must create a **Universal-Share** backup policy on the NetBackup primary server before using this feature.

To run a backup policy, you must add the backup key-value pairs and the `backup_selection` key-value pairs into the `.vpfs_special_control_config`. The `backup_selection` is only required if you want to backup special paths or directories for each backup. The `backup_selection` feature updates the backup selection of the policy each time the policy runs. If you use `backup_selection`, you are required to always provide the `backup_selection`.

Use the following for the backup key-value pair:

```
backup=<NetBackup policy name>:<backup schedue>:<backup client>
```

Use the following for the backup selection key-value pair:

```
backup_selection=<path1>:<path2>:<pathN>
```

Example for creating a backup from a workload:

```
(echo [vpfs]&& echo backup=
ushare-policy-01:full-backup-schedule:ushare-client
&& echo ingest_mode=off) >
<nfs_mount_point>/.vpfs_special_control_config
```

Example for creating a backup using a `backup_selection`:

```
(echo [vpfs]&& echo backup=
ushare-policy-01:full-backup-schedule:ushare-client && echo
backup_selection=
/mnt/vpfs_shares/usha/ushare/dir1:/mnt/vpfs_shares/usha/ushare/dir2
```

```
&& echo ingest_mode=off) >
<nfs_mount_point>/.vpfs_special_control_config
```

Universal share with disabled MSDP data volumes

Universal share supports multiple data volumes. The data and the metadata are balanced to all the configured data volume. In a multiple data volumes environment, universal share supports the ability to disable one or more than one of the data volumes for universal share. The data and the metadata for the new files are no longer written to a disabled data volume. However, universal share still may read data or metadata for the existing files. When object store is enabled for universal share, the local disk data will no longer write to the disabled data volume as well.

Note: The first data volume cannot be disabled for universal share.

To disable a data volume for universal share:

1. Use the `crcontrol` command to disable a data volume:

   ```
   crcontrol --dswriteoff [path]
   ```

2. Restart universal share:

   ```
   vpfs_mounts stop
   vpfs_mounts start
   ```

Universal Share WORM capability

The data in the universal share is not WORM capable if you access it from NFS or SMB. You can change or modify the data from NFS or SMB.

On a WORM storage server, use the Universal-Share policy to ingest the universal share into the NetBackup catalog, and make the catalog or the backup image WORM capable.

You can duplicate the image to another media server or replicate it to another NetBackup domain. You can also use the images to create an instant access universal share or restore data to the original universal share.

Universal share scale out

The universal share scale out feature enables you to use all the disk storage in a clustered environment. The universal share cluster distributes the file system operations across all the nodes or engines in the cluster environment, where each engine or node can access the data from a cluster. You can use any host name to

mount the universal share and access the data. This feature also provides the ability to control the access permission for the NFS/SMB server on the NetBackup Web UI. The data can be used directly on the object bucket or the local disk storage.

The regular single node universal share can coexist with the universal share cluster. It does not support an upgrade or conversion to a universal share cluster. If you upgrade to the current NetBackup version, there is no change in the regular single node universal share.

Supported platforms

- NetBackup Cloud Scale
- NetBackup Flex Scale

Limitations

- Some files may not be accessible if they are in the middle of a failover operation, which is triggered for an engine or node. The files are accessible after the failover operation is completed.
- The feature does not support multiple vpfsd instances.

Configure a universal share cluster

To configure a universal share cluster, you must first create a cluster and then mount the universal share cluster.

Create a universal share cluster

Prerequisite - Ensure that the TCP 10090 port is accessible between all the engines in the cluster.

Scenario 1: Direct data to a local disk storage

To direct data to a local storage

1. Create a Media Server Deduplication Pool.

   See “Configuring a storage server for a Media Server Deduplication Pool” on page 93.

2. Create a universal share.

   Select the type as **Cluster**.

   See “Create a universal share” on page 617.

Scenario 2: Direct data to a cloud bucket

To direct data to a cloud storage

1 Direct universal share data to an object store.

This step is only required for NetBackup Flex Scale.

See "Direct universal share data to object store" on page 639.

2 Create a universal share.

Select the type as **Cluster**.

See "Create a universal share" on page 617.

Mount a universal share cluster

Refer to the steps from the following section:

See "Mounting a universal share" on page 628.

Edit a universal share exported mount path

Refer to the steps from the following section:

See "View or edit a universal share" on page 626.

Deleting a universal share cluster

Refer to the steps from the following section:

See "Delete a universal share" on page 627.

Managing universal share services

You can manage universal shares using the following services:

- Veritas provisioning file system (VPFS)
- Variable-length deduplication (VLD) algorithm

Scan data volumes with the vpfs_stats utility

The `vpfs_stats` utility scans the data volumes and calculate the disk space usage consumed by vpfsd. It can help to calculate the data disk space which consumed by the data comes from vpfsd or universal share.

In the following cases, the disk space size will be shown as vpfsd space:

- Data has been written by vpfsd but backup images consume that data.
- All universal share/vpfsd data has been deleted, but there is still some universal share backup images.

The following example provides the format of the `vpfs_stats` utility:

```
vpfs_stats [options]...
```

Options:

- `--data-volume [--local] [--cloud]`
  Displays the VPFS data volume usage statistics. By default it displays the statistics for local storage usage and cloud storage usage. To display the statistics for local usage only, use the `--local` option. To display the statistics for cloud usage only, use `--cloud` option.
- `-help --help`
  Displays help text for the utility.

Here is an example of `vpfs_stats --data-volume`:

```
[root@hostname ~]# /usr/openv/pdde/vpfs/bin/vpfs_stats --data-volume

Start to scan data volume /msdp/vol1/data
Start to scan data volume /msdp/vol2
Finished scan data volume /msdp/vol1/data
Finished scan data volume /msdp/vol2

==================== Data volume summary ==============================
Number of data volume                       : 2

Data volume 1                               : /msdp/vol1/data
Number of data container groups             : 6
Number of vpfs data container groups        : 4
Vpfs data size                              : 2958014398 bytes(2.8GB)

Data volume 2                               : /msdp/vol2
Number of data container groups             : 4
Number of vpfs data container groups        : 4
Vpfs data size                              : 103 bytes(0.0MB)

All Data volumes
Number of data container groups             : 10
Number of vpfs data container groups        : 8
Vpfs data size                              : 2958014501 bytes(2.8GB)


==================== Vpfs cloud cache summary ========================
Number of data volume                       : 2

Data volume 1                               : /msdp/vol1/data
```

```
Number of data container groups             : 0
Number of vpfs data container groups        : 1
Vpfs data size                              : 4533165734 bytes(4.2GB)

Data volume 2                               : /msdp/vol2
Number of data container groups             : 0
Number of vpfs data container groups        : 1
Vpfs data size                              : 5129660901 bytes(4.8GB)

All Data volumes
Number of data container groups             : 0
Number of vpfs data container groups        : 2
Vpfs data size                              : 9662826635 bytes(9.0GB)


==================== Overall volume usage summary ====================
Number of data volume                       : 2

Data volume 1                               : /msdp/vol1/data
Number of data container groups             : 6
Number of vpfs data container groups        : 5
Vpfs data size                              : 2958014398 bytes(7.0GB)

Data volume 2                               : /msdp/vol2
Number of data container groups             : 4
Number of vpfs data container groups        : 5
Vpfs data size                              : 103 bytes(4.8GB)

All Data volumes
Total number of data container groups       : 10
Total number of vpfs data container groups  : 10
Total vpfs data size                        : 12620841136 bytes(11.8GB)
```

Changing the number of vpfsd instances

A universal share uses one vpfsd instance by default. In most cases, one instance is adequate. Increasing the number of vpfsd instances might improve universal share performance, although it also requires more CPU and memory. You can increase the number of vpfsd instances from 2 to up to 16 and distribute the shares cross all the vpfsd instances.

To change the number of vpfsd instances for universal shares

1. Stop NetBackup on the media server.

   ```
   systemctl stop netbackup
   ```

   Or

   ```
   /usr/openv/netbackup/bin/goodies/netbackup stop
   ```

2. Modify the number of vpfsd instances.

   Change the `numOfInstance` value in the `vpfsd_config.json` file. The value must be an integer between 2 and 16. For example:

   ```
   grep numOfInstance /msdp/vol1/etc/puredisk/vpfsd_config.json
   "numOfInstance": 2,
   ```

   BYO (build-your-own): `<storage path>/etc/puredisk/vpfsd_config.json`

   NetBackup Appliance and NetBackup Flex Scale: `/msdp/data/dp1/pdvol/etc/puredisk/vpfsd_config.json`

   NetBackup Flex: `/mnt/msdp/vol0/etc/puredisk/vpfsd_config.json`

3. Start NetBackup on the media server.

   ```
   systemctl start netbackup
   ```

   Or

   ```
   /usr/openv/netbackup/bin/goodies/netbackup start
   ```

Note: NetBackup 10.3 uses separate vpfsd instance for malware scanning, hence at least 1 vpfsd instance must be reserved. The vpfsd instance for malware scanning can be configured by changing the value of **numOfScanInstance**. The value must be an integer between 1 and 4, and **numOfScanInstance** must be less than **numOfInstance**.

Check the deduplication ratio for the universal share or a folder in the universal share

Check the deduplication ratio for a universal share:

```
/usr/openv/pdde/vpfs/bin/vpfs_metadump dedupe
/mnt/vpfs_shares/<share_dir>/<share_id>
```

Check the deduplication ratio for a universal share folder:

```
/usr/openv/pdde/vpfs/bin/vpfs_metadump dedupe
/mnt/vpfs_shares/<share_dir>/<share_id> <sub_dir>
```

Example usage and output:

```
                /usr/openv/pdde/vpfs/bin/vpfs_metadump dedupe
                /mnt/vpfs_shares/02b1/02b1e846-949f-5e55-8e39-e9900cd6a25e LT_0.1_20_1

File Name   File Size   Stored Size Overall Rate    Dedupe Rate Compress Rate
[INFO]: /LT_0.1_20_1/db_dump.1of14: 3043.42MB, 30.26MB, 99%, 93.31%, 85%
[INFO]: /LT_0.1_20_1/db_dump.2of14: 3043.42MB, 28.10MB, 99%, 93.94%, 84%
[INFO]: /LT_0.1_20_1/db_dump.3of14: 3045.02MB, 32.78MB, 98%, 92.82%, 85%
[INFO]: /LT_0.1_20_1/db_dump.4of14: 3044.93MB, 38.48MB, 98%, 91.44%, 85%
[INFO]: /LT_0.1_20_1/db_dump.5of14: 3044.93MB, 29.05MB, 99%, 93.78%, 84%
[INFO]: /LT_0.1_20_1/db_dump.6of14: 3044.93MB, 30.06MB, 99%, 93.45%, 84%
[INFO]: /LT_0.1_20_1/db_dump.9of14: 3043.42MB, 26.71MB, 99%, 94.27%, 84%
[INFO]: /LT_0.1_20_1/db_dump.8of14: 3043.42MB, 32.05MB, 98%, 93.07%, 84%
[INFO]: /LT_0.1_20_1/db_dump.10of14: 3043.42MB, 31.12MB, 98%, 93.36%, 84%
[INFO]: /LT_0.1_20_1/db_dump.12of14: 3044.93MB, 31.57MB, 98%, 93.13%, 84%
[INFO]: /LT_0.1_20_1/db_dump.11of14: 3044.93MB, 27.08MB, 99%, 94.23%, 84%
[INFO]: /LT_0.1_20_1/db_dump.7of14: 3043.42MB, 25.31MB, 99%, 94.65%, 84%
[INFO]: /LT_0.1_20_1/db_dump.13of14: 3044.93MB, 31.09MB, 98%, 93.33%, 84%
[INFO]: /LT_0.1_20_1/db_dump.14of14: 3044.93MB, 36.60MB, 98%, 91.79%, 85%
[INFO]: total size: 42620.06MB, stored size: 430.25MB, overall rate: 98.99%,
dedupe rate: 93.33%, compress rate:84%
    [0K, 8K): 0.0%
    [8K, 16K): 0.0%
    [16K, 24K): 0.7%
    [24K, 32K): 0.5%
    [32K, 40K): 98.8%
[INFO]: total SO: 1368688, average SO: 31K
```

Configuring the variable-length deduplication (VLD) algorithm for universal shares

The deduplication engine breaks the backup image into the segments and compares the segments to all of the segments that are stored in that deduplication node. Only the unique segments are sent to the NetBackup Deduplication Engine on the storage server. The Deduplication Engine writes the data to a Media Server Deduplication Pool.

See "About variable-length deduplication on NetBackup clients" on page 174.

NetBackup Deduplication Engine provides a couple of variable-length deduplication algorithm types. It brings better deduplication ratio if you use one variable-length deduplication algorithm for universal shares.

The variable-length deduplication algorithm is not enabled by default for universal shares, but the variable-length deduplication algorithm will be automatically enabled

for MSSQL and Sybase applications. Use the `vpfs_actions` command-line utility to manage the algorithm configuration.

To configure the variable-length deduplication algorithm for universal shares

1. Navigate to the following location on the media server:

   ```
   /usr/openv/pdde/vpfs/bin/
   ```

2. Run the following command to check the current configuration:

   ```
   /usr/openv/pdde/vpfs/bin/vpfs_actions -a tune --imageId <share_id>
   ```

   Sample output:

   ```
   segment_type: "vld"
   applications: [{"type": "vld", "sw_min": 16, "sw_max": 32}]
   status: 0
   ```

3. Configure the variable-length deduplication version.

   ```
   /usr/openv/pdde/vpfs/bin/vpfs_actions -a tune --imageId <share_id>
   --segment <VLD-version> --sw_min <sw_min> --sw_max <sw_max>
   ```

   Note: For the new environment where image backups do not exist in the storage, universal share automatically uses VLD v2 instead of VLD when you specify `-segment VLD` in the first-time configuration.

Option	Description
`imageId`	Unique identifier for the image.
`segment`	■ `alignment` Use fixed-length deduplication method. It's the default value. If set to `alignment`, `sw_min` and `sw_max` are not required. ■ `vld` The version one of variable length deduplication algorithm. ■ `vldv2` The version 2 of variable length deduplication algorithm. It is recommended to use this version as default. ■ `vldv3` Another version of variable length deduplication algorithm.
`sw_min`	The minimum segment size (KB) of the segmentation range (16 - 127). Suggested values are 16, 32, and 64.

Option	Description
`sw_max`	The maximum segment size (KB) of the segmentation range (17 - 128), this value must be greater than `sw_min`. Suggested values are 32, 64, and 128.

Using the marker file interface for universal share operations

A VPFS scheduler daemon provides a marker file interface to perform various universal share operations.

To configure the VPFS scheduler daemon

1. Enable the VPFS scheduler daemon.

   The VPFS scheduler is enabled automatically after one of its following features is enabled.

   - The universal share concurrent dump control
     See the topic To configure the universal share concurrent dump control for more information.
   - The automated universal share backup.
     See the topic To configure the automated universal share backup for more information.

2. The VPFS scheduler daemon starts automatically after at least one feature is enabled. To start VPFS scheduler manually, run the following command:

   ```
   /usr/openv/pdde/vpfs/bin/vpfs_sched start
   ```

3. Configure the following parameters of the VPFS scheduler daemon.

`VpfsScheduler.vpfsSchedInterval`	The interval time in seconds between each loop of operations.
`VpfsScheduler.vpfsSchedLogLevel`	The minimum logging level. Allowed values: error, warning, information, and debug.

To configure the universal share concurrent dump control

1 Enable the universal share concurrent dump control.

```
/usr/openv/pdde/vpfs/bin/vpfs_actions --action setVpfsConfigValue
--key enableShareDumpThrottling --value true
```

2 Configure the following parameters of universal share concurrent dump control.

`enableShareDumpThrottling`	Specify whether the share dump throttling feature is enabled or not. If false, no share concurrency is enforced.
`VpfsScheduler.maxConcurrent ActiveClientHostSharePairs`	The maximum “Client Host-Share Pairs” that can be active at once. When this limit is reached, any requested dumps that belong to the new “Client Host-Share Pair” are put into a queue. They are activated when other “Client Host-Share Pairs” are set to inactive. A higher value means that more dumps happen concurrently. The higher concurrent dumps may not yield a higher aggregate throughput and the concurrency needs to be tuned for a given system.
`VpfsScheduler.shareDumpsInactive Threshold`	The minimum time in seconds since the last IO operation in a share that the VPFS scheduler daemon waits. After the specified time, “Client Host-Share Pair” is inactive. For example, if this value is set to 300, and no IO operations have been detected in a share in at least 300 seconds, the “Client Host-Share Pair” is removed from the active list.
`VpfsScheduler.enableShareDump ThrottlingReporting`	Specify whether to log the history of movement of “Client Host-Share Pairs” within the throttle table. Movement includes “Client Host-Share Pairs” being queued, activated, and deactivated.

3 Copy dump scripts to the existing shares that are created before feature is enabled.

```
/usr/openv/pdde/vpfs/bin/vpfs_actions --action syncShareScript
```

4 Add the dump starts script to the start of the workload and use the `dump_end` script to the end of the workload.

For example:

- Windows batch dump scripts

```
\\hostname\ushare-smb1\.share-builtin-scripts\dump_start.bat
--dump-id <dump_id>

Existing workload dumping data to universal share

\\hostname\ushare-smb1\.share-builtin-scripts\dump_start.bat
--dump-id <dump_id>
```

- Linux bash dump scripts

```
/mnt/ushare_smb1/.share-builtin-scripts/dump_start.sh
--dump-id <dump_id>

Existing workload dumping data to universal share

/mnt/ushare_smb1/.share-builtin-scripts/dump_end.sh
--dump-id <dump_id>
```

Note: You must provide the `dump_id`. A randomly generated ID is recommended but it must be different from other dump scripts that are used in the same ClientHost-Share Pair. For SQL server, it is recommended to use the Job ID as the `dump_id`.

- View the throttling table for existing universal dumps.

```
/usr/openv/pdde/vpfs/bin/vpfs_actions -a dumpThrottleTable
```

To configure the automated universal share backup

1. Enable the automated universal share backup.

   ```
   /usr/openv/pdde/vpfs/bin/vpfs_actions --action setVpfsConfigValue
   --key VpfsScheduler.autoUShareBackupEnabled --value true
   ```

2. Configure the following parameters for automated universal share backup:

`VpfsScheduler.autoUShareBackupEnabled`	Enable the automated universal share backup feature.
`VpfsScheduler.autoUShareBackupCheckInterval`	Interval for checking touch files on universal share.
`VpfsScheduler.autoUShareBackupJobCheckInterval`	Interval for checking the status of running universal share backup job that is triggered by this feature.
`VpfsScheduler.autoUShareBackupPolicyFile`	The location of policy file that is configured in the file system.
`VpfsScheduler.autoUShareBackupOutputFile`	The location of an output file in the file system. The completed universal share backup job detail is written into the file.

3. Configure the universal share backup policy to be triggered by automated universal share backup.

   Configure a policy file that consists of a list of universal share policies that need to be triggered. The policy file location is a configurable path `VpfsScheduler.autoUShareBackupPolicyFile`, which can be configured using the `vpfs_actions` command.

   Each entry of the policy configuration consists of the policy name, client name, and the marker file location.

   For example,

   ```
   pl-ushare-nfs-1,host.domain.com,/mnt/vpfs_shares/
   usha/ushare-nfs-1/backup
   pl-ushare-nfs-2,host.domain.com,/mnt/vpfs_shares/usha/ushare
   -nfs-2/backup
   ```

   When the policy file is modified, the policy configuration is automatically reloaded.

4. Configure the backup schedule rotation.

The backup schedule is configured with a key-value pair where the key is the weekday (Monday - Sunday) and the value is the list of universal share backup type. The default configuration is FULL, INCR, CINC every day from Monday to Sunday.

The following are the supported backup types:

- FULL: Full backup
- INCR: Differential incremental backup
- CINC: Cumulative incremental backup

5 Create a marker file in the configured location in the universal share. After you configure the market file path in the policy file `VpfsScheduler.autoUShareBackupPolicyFile`, create marker file in designated location to trigger the backup.

The marker must be named with the following format:

`BACKUP_SUCCESS_<XXXX>_touch_<Date>` or `BACKUP_SUCCESS_<XXXX>_touch_<Date>_<Timestamp>`

Where,`<XXXX>` is optional and can be any string with no space.`<Date>` must be in the YYYYMMDD format.`<Timestamp>` should be a unique timestamp each time in a day such as HHmmSS.

The data and the timestamp are converted to a weekday. Then it is used to determine the backup schedule.

A universal share backup is triggered when it finds a new marker file that has a creation timestamp older than 5 minutes.

6 Verify that the backup is successful.

To check the status of the backup job, see NetBackup web UI. Once the backup is successful, the job information is also saved to the file that can be configured with `VpfsScheduler.autoUShareBackupOutputFile`.

The format of the output:

```
<job_id>,<schedule>,<state>,<status>,<total_size_kb>KB,<start_time>,
<end_time>,<client>,<policy>,<ushare_mount>
```

Troubleshooting issues related to universal shares

This section gives information about troubleshooting any problems you may encounter while you use universal shares.

Troubleshooting universal share configuration issues

How to troubleshoot a failed installation or configuration

To configure a universal share, ensure that instant access is enabled on the storage server. For more information about instant access, see the following guides:

- NetBackup for VMware Administrator's Guide
- NetBackup for Microsoft SQL Administrator's Guide

To ensure that instant access is enabled on the storage server

1. Log on to the storage server and run the following command (build your own (BYO) only):

   `/usr/openv/pdde/vpfs/bin/ia_byo_precheck.sh`

2. Review the pre-condition checking results and the configuration results:

   `/var/log/vps/ia_byo_precheck.log` (BYO only)

   `/usr/openv/pdde/vpfs/vpfs-config.log` (BYO and appliance configurations)

   In the following example, several required services are not running:

```
[root@rhelnbu06 ~]# /usr/openv/pdde/vpfs/bin/ia_byo_precheck.sh
Mon Apr 13 12:42:14 EDT 2020 Try to get storagepath
Mon Apr 13 12:42:14 EDT 2020 Storage ContentRouter config path is
   /msdp/etc/puredisk/contentrouter.cfg
Mon Apr 13 12:42:14 EDT 2020 Storagepath is /msdp
Mon Apr 13 12:42:14 EDT 2020 File system for partition /msdp is
   ext2/ext3
Mon Apr 13 12:42:14 EDT 2020 File system for partition /msdp/data
   is ext2/ext3
Mon Apr 13 12:42:14 EDT 2020 **** Hardware Virtualization not
   supported, Instant Access browse may be slow ****
Mon Apr 13 12:42:14 EDT 2020 **** system memory support 50 vpfs
   livemounts ****
Mon Apr 13 12:42:14 EDT 2020 **** nginx service required by
   Instant Access is not running ****
Mon Apr 13 12:42:14 EDT 2020 **** smb service required by
   Instant Access is not running ****
Mon Apr 13 12:42:14 EDT 2020 **** docker service required by
   VMware Instant Access is not running ****
```

3. Resolve the issues that are identified in the log. For example, restart any services that are required for instant access.

How to check for universal share capability

To ensure that the storage server has universal share capability

1. Make sure that the storage service is running NetBackup 8.3 or later.
2. Log on to the storage server and run the following command:

   ```
   nbdevquery -liststs -U
   ```

   Make sure that the `InstantAcess` flag is listed in the command's output.

   If the flag is not listed, see one of the guides mentioned above to enable instant access on the storage server.
3. Run the following command:

   ```
   nbdevconfig -getconfig -stype PureDisk -storage_server
   storage_server_name
   ```

   Make sure that the `UNIVERSAL_SHARE_STORAGE` flag is listed in the command's output.

   If the flag is not listed, create a universal share on the storage server:

   See "Create a universal share" on page 617.

How to start or to stop a universal share

A universal share can be started, restarted, or stopped with NetBackup services:

- Use the following command to start or restart a universal share:
  ```
  netbackup start
  ```
- Use the following command to stop universal share:
  ```
  netbackup stop
  ```

Whenever a universal share is created on the NetBackup web UI, a mount point is also created on the storage server.

For example:

```
[root@rsvlmvc01vm309 vpfs.mnt]# mount | grep vpfs
vpfsd on /mnt/vpfs type fuse.vpfsd (rw,nosuid,nodev,relatime,user_id=0,
   group_id=0,default_permissions,allow_other)
vpfsd on /mnt/vpfs_shares/aa7e/aa7e83e5-93e4-57ea-a4a8-81ddbf5f819e
   type fuse.vpfsd (rw,nosuid,nodev,relatime,user_id=0,group_id=0,
   default_permissions,allow_other)
```

In this example, `aa7e83e5-93e4-57ea-a4a8-81ddbf5f819e` is the universal share's ID. This ID is found on the details page of the universal share in the NetBackup web UI: On the left, click **Storage > Disk storage > Universal Share** and then select the universal share to view its details.

Logging and reporting for universal share VPFS instance

Additional reporting and logs are generated for VPFS operations. You can use this information for performance analysis and troubleshooting. VPFS write-statistics are saved into the vpfsd history in JSON format.

This file is stored at `<msdp_vol>/history/vpfsd-report` location in `vpfsX_YYYY-MM-DD` format. For example, /msdp/vol/history/vpfsd-report/vpfs0_2023-05-01

Sample file:

```
{"timestamp":1682906052.752,"threadId":140498966533888,"UID":"ushare
-1","UpdateCounter:":4846,"SegmentNumber":39148,"DedupeRate":0.00,"AvgDedupe
TimePerSegment(ms)":0.534,"AvgSegmentSize":128792.79,"AvgWriteTimePerSegment(
ms)":0.003,"TotalWriteSize":5041984256}
```

To configure the logging and reporting for VPSF instance

1. Export the history report to a CVS file:

   ```
   /usr/openv/pdde/vpfs/bin/vpfs_actions --action getVpfsHistoryStat
   --exportPath <export-path> [--targetDate <date> | --targetDateFrom
   <start-date> --targetDateTo <end-date>]
   ```

2. Modify the reporting interval:

   ```
   /usr/openv/pdde/vpfs/bin/vpfs_actions --action setVpfsConfigValue
   --key vpfsdReportInterval --value <newInterval>
   ```

   The reporting interval is 3600 seconds by default. Set it to 0 to disable logging and reporting for VPSF instance.

Vpfsd logs for file system operations in universal shares

Vpfsd log reporting enables logging of the file system operations that took a longer time than expected. There are two configurations to enable vpfsd log reporting. Both configurations are disabled by default.

To enable vpfsd logging for the file system operations

1. Enable logging for the file system operations:

   ```
   /usr/openv/pdde/vpfs/bin/vpfs_actions --action setVpfsConfigValue
   --key enableVpfsdLogReporting --value true
   ```

   The vpfsd logging is enabled for the following file system operations:

   - `getattr`
   - `rename`

- open
- read
- write
- fsync
- truncate
- getDataCacheNodeForWrite

2 Enable logging for the extended file system operations:

```
/usr/openv/pdde/vpfs/bin/vpfs_actions --action setVpfsConfigValue
--key enableVpfsdLogReportingExtended --value true
```

The vpfsd logging is enabled for all the file system operations.

3 Search the messages in the file:

```
cat /msdp/vol/log/vpfs/vpfsd/vpfs0_vpfsd.log | grep
"WARNING.*exceeded configured threshold"
```

4 (Optional) Change the logging threshold:

```
/usr/openv/pdde/vpfs/bin/vpfs_actions --action setVpfsConfigValue
--key logReportWriteTimeThreshold --value
```

Notify events to primary server with the vpfsd service

The `vpfsd` service sends critical events to the NetBackup primary server when it detects an issue, such as file corruption, checkpoint failure, and no NetBackup policy that is configured to protect a universal share.

In case of file corruption events, the universal share sends the event when it detects the following:

- Metadata of the file is corrupted or inconsistent.
- Data container is corrupted or lost.

For example:

```
File path: <Corrupted file path>

Share name: <Universal share name>

Storage server: <The storage server for universal share>

Share type: VPFS
```

Checkpoint failures can affect data consistency and an event is triggered if the checkpoint repeatedly fails.

It is recommended that you configure the NetBackup universal-share policy to protect the universal share. An event is sent to the NetBackup primary server if the universal share is not protected or there if there is no successful backup in a specified period of time (last 24 hours by default).

Data integrity check and recovery using the vpfsd service

The `vpfsd` service performs a data integrity check in the background according to the data changes. An event is sent to the NetBackup primary when file corruption is detected, which includes the metadata and the data corruption (automatic process).

The corrupted file is moved to the following location:

```
/mnt/vpfs_share/<share dir>/<share ID>/.vpfs_corrupted
```

A syslog and a history log are created for the file. The backup of the universal share skips the files that have an issue.

Apart from the automatic process, you can use the `vpfsck` command to manually verify and detect the file corruption, and the corrupted file is moved to the following location:

```
/mnt/vpfs_share/<share dir>/<share ID>/.vpfs_corrupted
```

Only local VPFS is supported.

For example:

```
--verify_data  --share_id <share id>
 [--t <file creation time older than (int) hours>]
 --check_so    --share_id <share id>
 [--so <so fp, find files that contain this so>]
 [--file_path <check single file or folder,
 path relative to /mnt/vpfs/<share_id<0:4>>/<share_id>>]
 --verbose  optional
```

All the options after `--share_id` are optional.

The process is logged at `<storage>/log/vpfs/vpfsd/vpfsck.log` and the option `--verbose` enables the debug log for `vpfsck`.

Chapter 16

Configuring isolated recovery environment (IRE)

This chapter includes the following topics:

- Requirements
- Configuring the network isolation
- Configuring an isolated recovery environment using the web UI
- Configuring an isolated recovery environment using the command line
- Replicating the backup images from the IRE domain to the production domain

Requirements

Following are the requirements to configure isolated recovery environment (IRE) in a Pull model:

- WORM storage server: 17.0 or later (on Flex Appliance: 2.1.1 or later)
- NetBackup BYO media server: 10.1 or later
- NetBackup Flex Scale: 3.2 or later
- Access Appliance: 8.2 or later
- Flex media server: 11.1 or later

Table 16-1 lists the supported configuration for MSDP source and targets for isolated recovery environment.

Table 16-1 Supported configuration for MSDP source and targets

Source	Target	Deployment models	NetBackup version	WORM version	Replication - OptDup model
MSDP, WORM	WORM	Flex 2.1	10.0	16.0	Push model
MSDP, WORM	MSDP, WORM	BYO, Flex 2.1.1	10.1	17.0	Pull model
MSDP, WORM	MSDP, WORM	BYO, Flex	10.1.1	17.1	Pull model with IPv6 + mixed CA support for IRE hosts
MSDP, WORM, MSDP Scaleout	MSDP, WORM, MSDP Scaleout	BYO, Flex, Flex Scale	10.2	18.0	Pull model
MSDP, WORM, MSDP Scaleout	MSDP, WORM, MSDP Scaleout	BYO, Flex, Flex Scale	10.3	19.0	Web UI

Note: For NetBackup 10.0 and WORM 16.0, download the hotfix `VRTSflex-HF3-2.1.0-0.x86_64.rpm` for the Flex Appliance in the IRE and NetBackup EEB `VRTSflex-msdp_EEB_ET4067891-16.0-3.x86_64.rpm` for the WORM storage server application from the Download Center.

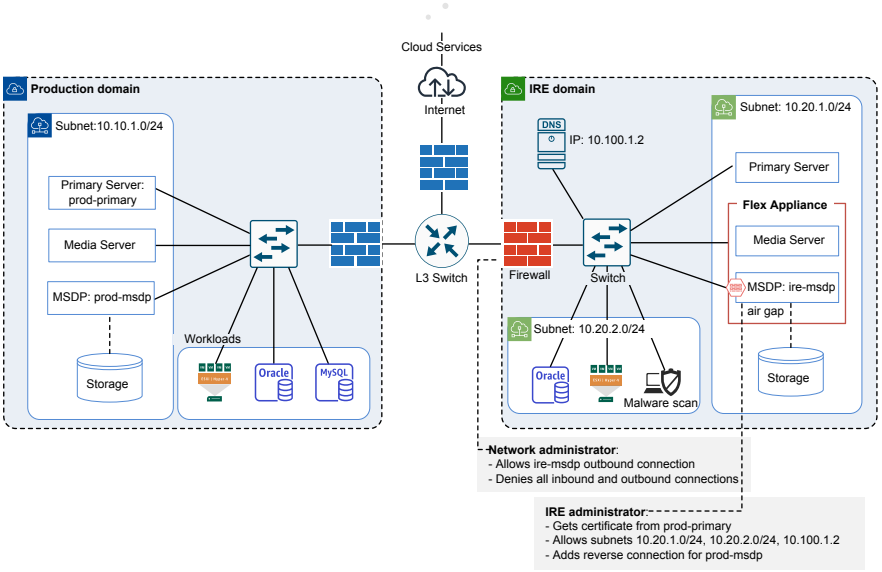
Note: Both the source and the target domain must meet the minimum software version requirements.

In the NetBackup releases prior to 10.3.0.1, the isolated recovery environment feature is supported on Red Hat Enterprise Linux BYO only. From NetBackup 10.3.0.1, the isolated recovery environment is supported on SUSE Linux Enterprise Server BYO also.

Configuring the network isolation

To support the AIR for IRE, network communication from the IRE MSDP storage server to the production MSDP storage server is required. IRE MSDP storage server initiates the network connection. IRE with AIR works even when the production MSDP server does not have network access to the IRE MSDP server.

Figure 16-1 Network isolation example



Configure network isolation in the firewall

Configure the firewall at the IRE domain to deny all the inbound and the outbound connections. It helps to protect all the hosts in IRE domain from the cyberattacks. You must allow the IRE MSDP server outbound connection. For IRE replication, the IRE MSDP server must have network access to the production MSDP server through the ports 10082 and 10102. The IRE MSDP server also must have network access to the production primary server using the port 1556.

If you cannot allow unidirectional network access (allow only outbound connection) on the firewall, you can allow bidirectional network for the IRE MSDP server. IRE air gap in the IRE MSDP server still denies all the inbound connections.

Configure IRE air gap on IRE MSDP server

Air gap in IRE MSDP server does the following:

- Allows the network connections from servers in IRE domain. Connectivity between MSDP server and NetBackup primary or media servers is required to make the MSDP server functional.

  Add the subnets or IP addresses of the IRE domain to the allowed subnet list. The IP addresses in the subnet list have direct network access to the MSDP server.

  For example, for Flex WORM use the following command:

```
setting ire-network-control allow-subnets
subnets=<subnet1>,<subnet2>,<ip address>,etc
```

Note: The list must have at least the primary server, the media servers, and the DNS server in IRE domain.

Do not add subnets or IP addresses from the domains outside the IRE domain.

- Enables a unidirectional network access (allow outbound connection from IRE MSDP server to the other domains) in IRE air gap window. By default, the window is 24 hours per day.

All the inbound connections that are not in the allowed subnet list are denied.

Configuring an isolated recovery environment using the web UI

Perform the following steps to configure an isolated recovery environment using the NetBackup web UI.

Note: You can also configure and manage an IRE from the deduplication shell. See the *NetBackup Deduplication Guide* for more details.

IRE web UI relies on Storage Platform Web Service (**spws**) on the IRE MSDP storage server. If the IRE MSDP storage server runs on a BYO media server, ensure that the **spws** service is configured and running. Ensure that NGINX is installed and started before you configure **spws** service.

See "Storage Platform Web Service (spws) does not start" on page 787. to configure **spws** service.

Table 16-2 IRE configuration using the NetBackup web UI

Step	Task	Description
1.	Configure allowed subnets to allow only the hosts in the subnets to access the storage server.	See "Configuring the allowed subnets" on page 679.
2.	Configure reverse connections to support replicating backup images from storage servers outside the isolated recovery environment.	See "Configuring the reverse connections" on page 679.

Table 16-2 IRE configuration using the NetBackup web UI *(continued)*

Step	Task	Description
3.	(Optional) Configure reverse replication schedule to allow network activities in a specific window.	See "Configuring the reverse replication schedule" on page 681.
4.	Configure SLP for replicating backup images from a production environment.	See "Adding a replication operation to SLP at the production primary server" on page 681.

Configuring the allowed subnets

Allowed subnets are like a firewall. Any hosts that are not in the allowed subnets has no access to the IRE MSDP server. Ensure that the IRE primary server is in the allowed subnets, otherwise you lose the control to the IRE MSDP server from the web UI. The computer you use to configure the IRE also must be in the allowed subnets.

Note: For Flex Scale, the allowed subnets protect the entire cluster including the nodes, the NetBackup servers, and MSDP engines. Ensure that all the subnets, which need to access the cluster are in the allowed subnets.

To configure the allowed subnets

1. On the left, click **Storage > Disk storage**.
2. Click the **Storage servers** tab.
3. Click on the MSDP storage server that you want to configure.
4. Under **Isolated Recovery Environment** > **Allowed subnets**, click **Add subnet**.
5. Select **IPv4** or **IPv6** and type the IP address of the subnet and click **Add to list**.
6. Add all the subnets in the IRE domain that are required to access the MSDP server and click **Save**.

Configuring the reverse connections

Before you add reverse connections from the IRE storage server to production storage server, ensure that NBCA or ECA are configured on the IRE storage server for the production domain.

For BYO media server: See the step 1 of the *Configuring A.I.R. for replicating backup images from production environment to IRE BYO environment* topic.

See "Configuring A.I.R. for replicating backup images from production environment to IRE BYO environment" on page 692.

For WORM storage server, Flex Scale, and Access Appliance: See the step 2 of the *Configuring data transmission between a production environment and an IRE WORM storage server* topic.

See "Configuring data transmission between a production environment and an IRE WORM storage server" on page 702.

To configure the reverse connections

1. On the left, click **Storage > Disk storage**.
2. Click the **Storage servers** tab.
3. Click on the MSDP storage server that you want to configure.
4. Under **Isolated Recovery Environment** > **Reverse connections**, click **Add reverse connection**.
5. On the **Add reverse connection** page, provide the production primary server name.
6. Select the existing login credentials or add new credentials and click **Next**.
   - **Select existing credentials**: Select the existing credentials.
   - **Add a new credential**: Add a new credential for the production primary server. Under **Credential type**, select **Username Password authentication** or **Use API key**.

     Note: The user of the production primary server needs privileges in the **default IRE SLP Administrator** role.

7. Click **Connect**.
8. On the next page, select **Remote MSDP storage server**.

   You can select an MSDP storage server from the production domain. If the MSDP storage server has multiple network interfaces configured and you want the reverse connection, use another interface rather than the storage server name. You can type the FQDN of the network interface for the production MSDP storage server.

9 In the **Local interface** field, provide the local storage server interface name for data transmission.

If the IRE MSDP server has multiple interfaces and you want the IRE MSDP server to use a specific interface to connect to the production MSDP storage server, type the FQDN of the network interface for the IRE MSDP storage server.

If nothing is specified in **Local interface** field, IRE MSDP server uses the default network interface to connect to the production storage server.

10 Click **Add**.

A reverse connection is configured from the IRE MSDP server to the production MSDP server.

Configuring the reverse replication schedule

By default, an IRE MSDP storage server allows reverse connections to production storage server in a big window (24x7). For security purpose, administrator may want the reverse connections to be created in a small window.

To configure the allowed subnets

1 On the left, click **Storage > Disk storage**.

2 Click the **Storage servers** tab.

3 Click on the MSDP storage server that you want to configure.

4 Under **Isolated Recovery Environment** > **Reverse replication schedule**, click **Configure schedule**.

5 Configure the window for each weekday to allow reverse connections. Click the **Reset to default 24/7schedule** to restore the window to the default.

6 Click **Save**.

Adding a replication operation to SLP at the production primary server

To configure the reverse connections

1 On the left, click **Storage > Disk storage**.

2 Click the **Storage servers** tab.

3 Click on the MSDP storage server that you want to configure.

4 Under **Isolated Recovery Environment**, click **Modify SLP on the remote primary server**.

5 On the **Modify SLP on the remote primary server** page, provide the production primary server name.

6 Select the existing login credentials or add new credentials and click **Next**.

- **Select existing credentials**: Select the existing credentials.
- A**dd a new credential**: Add a new credential for the production primary server. Under **Credential type**, select **Username Password authentication** or **Use API key**.

Note: The user of the production primary server needs privileges in the **default IRE SLP Administrator** role.

---

7 Click **Connect**.

8 Select the SLP that you want to add a replication operation to the IRE MSDP storage server and click **Next**.

9 Select an operation that you want to replicate to IRE MSDP storage server after the operation and click **Next**.

10 Select an SLP of the IRE domain for image import after replication completed.

11 On the **Window** tab, configure SLP window for the replication operation. Create a new SLP window or select an existing SLP window.

When you adjust the SLP window, ensure that the SLP window is covered by IRE schedule. If a replication is triggered outside the IRE schedule, reverse connection does not happen, and the replication job fails.

The **Synchronize with the reverse connection schedule** helps to replace the current SLP window with the IRE Schedule. You can adjust the SLP window based on the IRE Schedule.

The date and time that is shown on the page are based on the time zone of IRE primary server. If the production primary server and the IRE primary server are in different time zones, the time difference is calculated and the SLP window for the production primary server is converted automatically.

Click **Finish**.

12 Click **Save**.

All the configurations including MSDP storage server replication target, SLP window, and replication operation in the SLP are applied to the production primary server.

Configuring an isolated recovery environment using the command line

See the following topics to configure an isolated recovery environment using the command line:

Table 16-3 IRE configuration using the command line

Task	Description
Configure an isolated recovery environment on a NetBackup BYO media server.	See "Configuring an isolated recovery environment on a NetBackup BYO media server" on page 684.
Manage an isolated recovery environment on a NetBackup BYO media server.	See "Managing an isolated recovery environment on a NetBackup BYO media server" on page 689.
Configure an AIR for replicating backup images from production environment to IRE BYO environment.	See "Configuring A.I.R. for replicating backup images from production environment to IRE BYO environment" on page 692.
Configure an isolated recovery environment on a WORM storage server.	See "Configuring an isolated recovery environment on a WORM storage server" on page 696.
Manage an isolated recovery environment on a WORM storage server.	See "Managing an isolated recovery environment on a WORM storage server" on page 699.
Configure the data transmission between a production environment and an IRE WORM storage server.	See "Configuring data transmission between a production environment and an IRE WORM storage server" on page 702.

Note: Configuring an isolated recovery environment using the NetBackup web UI or command line is the same for both Flex media servers and BYO media servers.

Using the API Keys to support MFA configuration

You can use API keys when accessing the production and IRE primary servers. API keys provide an alternative to traditional password-based authentication, especially in environments where multi-factor authentication (MFA) is enabled. Unlike passwords, API keys bypass the need for an MFA one-time password (OTP).

The `show_slp_window`, `sync_ire_window`, and `add_replication_op` commands that require authentication to the primary server prompt for the admin user's

password if an API key is not specified. However, these commands do not support MFA OTP. If MFA is enabled on the primary server, password-based authentication will be denied, preventing access. This limitation makes API keys essential for environments that have MFA enabled.

Use the `production_use_apikey` and `ire_use_apikey` parameters in the commands to authenticate with API keys instead of passwords. The `production_use_apikey` parameter is used for accessing the production primary server, while the `ire_use_apikey` parameter is used for the IRE primary server. Including these parameters in the command ensures seamless authentication without requiring an OTP.

To use these parameters, administrators must ensure that the API key is associated with a user account that has the necessary Role-Based Access Control (RBAC) permissions. Specifically, the admin user for the API key must have at least the Default IRE SLP Administrator role. For information on managing API keys, refer to the *NetBackup Web UI Administrator's Guide*.

Configuring an isolated recovery environment on a NetBackup BYO media server

You can configure an isolated recovery environment (IRE) on a NetBackup BYO media server to create an air gap between your production environment and a copy of the protected data. The air gap restricts network access to the IRE environment all the time. This feature helps to protect against ransomware and malware. To configure an IRE, you need a production NetBackup environment and a NetBackup IRE environment with MSDP server configured in a BYO Media server. The production environment does not require any additional steps for this feature.

Use the following procedure to configure an IRE on a BYO media server.

To configure an IRE on a BYO media server

1. Note that this procedure applies only to NetBackup 10.1 and later.

   Log in to the media server.

2. This step is optional. Use this step in any of the following conditions:

   - You want to enable IRE on an existing system.
   - AIR SLP is already configured.
   - You want to configure the IRE schedule in step 4 based on the existing SLP window.

   Run the following command to show the SLP windows for replication from the primary server to the MSDP storage on the media server:

```
/usr/openv/pdde/shell/bin/show_slp_windows
--production_primary_server production primary server name
--production_primary_server_username production primary server
username --ire_primary_server target primary server name
--ire_primary_server_username target primary server username
[--production_use_apikey] [--ire_use_apikey]
```

Where:

- The *production primary server name* is the fully qualified domain name (FQDN) of the primary server in your production environment.
- The *production primary server username* is the username of a NetBackup user with permission to list SLPs and SLP windows in the production environment.
  The *production primary server username* must be in `domain_name\user_name` format on Windows.
- The *target primary server name* is the FQDN of the primary server in the IRE. Use the same hostname that you used to configure the SLPs in the production environment.
- The *target primary server username* is the username of a NetBackup user with permission to list the SLPs and storage units in the IRE environment.
  The *target primary server username* must be in `domain_name\user_name` format on Windows.

For example:

```
production_primary_server=examplePrimary.domain.com
production_primary_server_username=appadmin
ire_primary_server=exampleIREPrimary.domain.com
ire_primary_server_username=appadmin
```

The following is an example output of the command:

```
EveryDayAtNoon: SLPs: SLP1 Sunday start: 12:00:00 duration: 00:59:59
Monday start: 12:00:00 duration: 00:59:59 Tuesday start: 12:00:00
duration: 00:59:59 Wednesday start: 12:00:00 duration: 00:59:59
Thursday start: 12:00:00 duration: 00:59:59 Friday start: 12:00:00
duration: 00:59:59 Saturday start: 12:00:00 duration: 00:59:59
WeeklyWindow: SLPs: SLP2 Sunday start: 10:00:00 duration: 01:59:59
Monday NONE Tuesday NONE Wednesday NONE Thursday NONE Friday NONE
Saturday start: 10:00:00 duration: 01:59:59
```

This example shows two SLP windows:

- A daily window for one hour starting at noon.

- A weekly window for 2 hours starting at 10 A.M.

Note: If an SLP window is greater than 24 hours, the `show-slp-windows` may display the duration incorrectly.

---

3 Based on the output for your environment, determine a daily schedule that accommodates the SLP windows and take note of it. In the previous example, a daily schedule from 10 A.M. to 12:00 P.M. accommodates both SLP windows.

The start times in the output of this command are in the IRE server's time zone.

Note: If the time zone of the production primary server is changed, you must restart the NetBackup services.

---

4 Run the following command to configure the subnets and IP addresses that are allowed to access the media server:

```
/usr/openv/pdde/shell/bin/ire_network_control allow-subnets
--subnets CIDR subnets or IP addresses
```

Where the *CIDR subnets or IP addresses* field is a comma-separated list of the allowed IP addresses and subnets in CIDR notation.

For example:

```
/usr/openv/pdde/shell/bin/ire_network_control allow-subnets
--subnets 10.10.100.200,10.80.40.0/20
```

Note: The IRE primary server, the IRE media servers, and the DNS server for the IRE environment must be included in the allowed list. If all these servers are in the same subnet, only the subnet is required to be in the allowed list.

Note: If your network environment is dual stack, ensure that both IPv4 and IPv6 subnets and IP addresses of the IRE domain are configured in allowed subnets. For example, if you specify only IPv6 subnets in the allowed subnet, all the IPv4 addresses are not allowed to access the IRE storage server.

---

5 Run the following command to set the daily air gap schedule:

```
/usr/openv/pdde/shell/bin/ire_network_control set-schedule
--start_time time --duration duration [--weekday 0-6]
```

`weekday` is optional. It starts from Sunday. You can configure different network and open or close window for a specific weekday. If it is not specified, the IRE schedule is the same on each day.

For example:

```
/usr/openv/pdde/shell/bin/ire_network_control set-schedule
--start_time 10:00:00 --duration 03:00:00
```

Note: The SLP replication window on the production domain must be configured to be open at the same time as the IRE schedule. The IRE schedule window can be different for weekdays. You can configure a window for a specific weekday.

For example:

```
/usr/openv/pdde/shell/bin/ire_network_control set-schedule
--start_time 11:00:00 --duration 10:00:00 --weekday 0
```

Note: If the production and the IRE environments are in different time zones, the schedule must begin only once per day in both time zones.

For example, if one environment is in the Asia/Kolkata time zone and the other is in the America/New_York time zone, the following schedule in Kolkata is not supported: Tuesday start time 22:00:00 and Wednesday start time 03:00:00. When these start times are converted to the New York time zone, they become Tuesday start time 12:30:00 and Tuesday start time 17:30:00, which is not supported.

Note: If you want to open air gap network for 24 hours on all days, you do not need to configure IRE schedule. However, the IRE media server restricts the network access from the hosts that are not configured in the subnets that the air gap allows.

Managing an isolated recovery environment on a NetBackup BYO media server

Once you have configured an isolated recovery environment on a NetBackup BYO media server, you can manage it from the media server.

Use the following commands:

To view the SLP windows from the primary server to the WORM instance:

- `/usr/openv/pdde/shell/bin/show_slp_windows --production_primary_server` *`production primary server name`* `--production_primary_server_username` *`production primary server username`* `--ire_primary_server` *`target primary server name`* `--ire_primary_server_username` *`target primary server username`* `[--production_use_apikey] [--ire_use_apikey]`

  Where:

  - The *production primary server name* is the fully qualified domain name (FQDN) of the primary server in your production environment.
  - The *production primary server username* is the username of a NetBackup user with permission to list SLPs and SLP windows in the production environment.
    The *production primary server username* must be in `domain_name\user_name` format on Windows.
  - The *target primary server name* is the FQDN of the primary server in the IRE. Use the same hostname that you used to configure the SLPs in the production environment.
  - The *target primary server username* is the username of a NetBackup user with permission to list the SLPs and storage units in the IRE environment. For example:
    The *target primary server username* must be in `domain_name\user_name` format on Windows.

    ```
    production_primary_server=examplePrimary.domain.com
    production_primary_server_username=appadmin
    ire_primary_server=exampleIREPrimary.domain.com
    ire_primary_server_username=appadmin
    ```

To view the allowed IP addresses and subnets

- Run the following command:

  ```
  /usr/openv/pdde/shell/bin/ire_network_control show-allows
  ```

To add IP addresses and subnets to the allowed list

- Run the following command:

  ```
  /usr/openv/pdde/shell/bin/ire_network_control allow-subnets
  --subnets CIDR subnets or IP addresses
  ```

  The *CIDR subnets or IP addresses* field is a comma-separated list of the allowed IP addresses and subnets, in CIDR notation.

  For example:

  ```
  /usr/openv/pdde/shell/bin/ire_network_control allow-subnets
  --subnets 10.60.120.208,10.74.48.0/20
  ```

  Note: The IRE primary server, the IRE media servers, and the DNS server for the IRE environment must be included in the allowed list. If all these servers are in the same subnet, only the subnet is required to be in the allowed list.

  Note: If your network environment is dual stack, ensure that both IPv4 and IPv6 subnets and IP addresses of the IRE domain are configured in allowed subnets. For example, if you specify only IPv6 subnets in the allowed subnet, all the IPv4 addresses are not allowed to access the IRE storage server.

To remove the IP addresses and subnets from the allowed list

- Run the following command:

  ```
  /usr/openv/pdde/shell/bin/ire_network_control allow-subnets
  --subnets
  ```

To view the daily air gap schedule

- Run the following command:

  ```
  /usr/openv/pdde/shell/bin/ire_network_control show-schedule
  ```

To change the air gap schedule

- Run the following command:

  ```
  /usr/openv/pdde/shell/bin/ire_network_control set-schedule
  --start_time time --duration duration [--weekday weekday in 0-6]
  ```

  For example:

  ```
  /usr/openv/pdde/shell/bin/ire_network_control set-schedule
  --start_time 10:00:00 --duration 03:00:00
  ```

  Note: The SLP replication window on the production domain must be configured to be open at the same time as the IRE schedule.

To stop the air gap schedule

- Run the following command:

  ```
  /usr/openv/pdde/shell/bin/ire_network_control delete-schedule
  [--weekday weekday in 0-6]
  ```

  Note: You can delete an IRE window for a specific weekday.

To view the current network status and check whether the external network is open or closed

- Run the following command:

  ```
  /usr/openv/pdde/shell/bin/ire_network_control
  external-network-status
  ```

To manually open the external network

- Run the following command:

  ```
  /usr/openv/pdde/shell/bin/ire_network_control
  external-network-open
  ```

To manually close the external network and resume the air gap schedule

- Run the following command:

  ```
  /usr/openv/pdde/shell/bin/ire_network_control resume-schedule
  ```

To add MSDP reverse connection

- Run the following command:

```
/usr/openv/pdde/shell/bin/ire_network_control reverse-connection
--add source msdp server [--remote_primary source primary server]
[--local_addr local msdp server]
```

To remove MSDP reverse connection

- Run the following command:

```
/usr/openv/pdde/shell/bin/ire_network_control reverse-connection
--remove source msdp server
```

To list configured MSDP reverse connections

- Run the following command:

```
/usr/openv/pdde/shell/bin/ire_network_control reverse-connection
--list
```

To validate if a specific reverse connection works

- Run the following command:

```
/usr/openv/pdde/shell/bin/ire_network_control reverse-connection
--validate source msdp server
```

Configuring A.I.R. for replicating backup images from production environment to IRE BYO environment

Once IRE configuration is completed, the production NetBackup hosts are no longer able to access the IRE MSDP storage server. You need to enable MSDP reverse connection to allow the data transmission between the production MSDP server and the IRE MSDP server.

Note: A.I.R. configuration operations can be performed when the external network is open by IRE air gap. All the given operations are performed on the IRE MSDP server.

Prerequisites

Before you configure A.I.R. for replicating backup images from production environment to IRE BYO environment, ensure the following:

- In the case of NetBackup certificate authority (CA), get the CA certificate and host certificate for the IRE MSDP storage server from the production primary server.

- Create a token on the production primary server.

To configure A.I.R. for replicating backup images from production environment to IRE BYO environment

1. Run the following commands:
   - NetBackup certificate:
     ```
     /usr/openv/netbackup/bin/nbcertcmd -getCACertificate -server
     <production primary server>
     /usr/openv/netbackup/bin/nbcertcmd -getCertificate -server
     <production primary server> -token <token>
     ```
   - External certificate:
     ```
     /usr/openv/netbackup/bin/nbcertcmd -enrollCertificate -server
     <production primary server>
     ```
2. Run the following command to enable MSDP reverse connection.
   ```
   /usr/openv/pdde/shell/bin/ire_network_control reverse_connection
   --add production msdp server
   ```

3 This step is not required if you have not configured any IRE schedule. That is because if the IRE schedule is not configured, MSDP reverse connection is enabled for 24 hours on all days. The production primary server can configure the SLP replication operation with any SLP window.

Once the MSDP reverse connection is configured, copy the IRE schedule to the NetBackup production domain as an SLP window. Use the following command:

```
/usr/openv/pdde/shell/bin/sync_ire_window
--production_primary_server production primary server name
--production_primary_server_username production primary server
username [--slp_window_name slp_window_name ]
[--production_use_apikey]
```

Where:

The *production primary server name* is the fully qualified domain name (FQDN) of the primary server in your production environment.

The *production primary server username* is the username of a NetBackup user with permission to list SLPs and SLP windows in the production environment.

The *production primary server username* must be in `domain_name\user_name` format on Windows.

The *slp_window_name* is the name of the SLP window to be synced with the IRE window. It is an optional parameter. If the SLP window is not specified, an SLP window with the name `IRE_DEFAULT_WINDOW` is created on the production primary server.

4 You can then add the IRE MSDP storage server as a replication target of the production NetBackup domain. Then add the replication operation to an existing SLP to replicate from production NetBackup domain to IRE MSDP storage server using the following command:

```
/usr/openv/pdde/shell/bin/add_replication_op
--production_primary_server production primary server name
--production_primary_server_username production primary server
username --source_slp_name source slp name
--target_import_slp_name target import slp name
--production_storage_server production storage server name
--ire_primary_server_username ire primary server username
--target_storage_server target storage server name
--target_storage_server_username target storage server username
--production_storage_unit msdp storage unit name used in source
SLP [--slp_window_name slp window name] [--production_use_apikey]
[--ire_use_apikey]
```

Where:

The *production primary server name* is the fully qualified domain name (FQDN) of the primary server in your production environment.

The *production primary server username* is the username of a NetBackup user with permission to list SLPs and SLP windows in the production environment.

The *production primary server username* must be in `domain_name\user_name` format on Windows.

The *production storage server name* is the fully qualified domain name (FQDN) of the production storage server in your production environment.

The *ire primary server username* is the username for administrator user of IRE primary server.

The *ire primary server username* must be in `domain_name\user_name` format on Windows.

The *source slp name* is the SLP name on the production primary server against which a replication operation is added.

The *target import slp name* is the import SLP name from IRE primary server.

The *target storage server name* is the fully qualified domain name (FQDN) of the target MSDP storage server.

The *target storage server username* is the username of the target MSDP storage server.

The *slp_window_name* is the name of the SLP window that is synced with the IRE window. Alternatively, it is created on the production primary server before the operation. It is an optional parameter. If the SLP window is not specified, an SLP window with the name `IRE_DEFAULT_WINDOW` is used that must be created using the `sync_ire_window` command before the operation.

The *production_storage_unit* is the storage unit name of type PureDisk used in source SLP.

Note: The source SLP and target import SLP need to be created before the operation.

Configuring an isolated recovery environment on a WORM storage server

You can configure an isolated recovery environment (IRE) on a WORM storage server to create an air gap between your production environment and a copy of the protected data. The air gap restricts network access to the data except during the timeframe when data replication occurs. This feature helps to protect against ransomware and malware.

To configure the IRE, you need a production NetBackup environment and a target WORM storage server on a supported Cohesity appliance. Check the appliance documentation for compatibility.

The production environment does not require any additional steps for this feature. Use the following procedure to configure an IRE on the target WORM storage server from the deduplication shell.

To configure an IRE

1 If A.I.R. is not configured on the production domain, continue to the next step.

If A.I.R. is already configured on the production domain, log in to the deduplication shell as the **msdpadm** user. Run the following command to show the SLP windows for replication from the primary server to the WORM server.

```
setting ire-network-control show-slp-windows
production_primary_server=<production domain>
production_primary_server_username=<production username>
ire_primary_server=<IRE domain> ire_primary_server_username=<IRE
username> [production_use_apikey] [ire_use_apikey]
```

Where:

- *<production domain>* is the fully qualified domain name (FQDN) of the primary server in your production environment.
- *<production username>* is the username of a NetBackup user with permission to list SLPs and SLP windows in the production environment. For Windows users, enter the username in the format ***<domain name>\<username>***. For other users, enter the username only.
- *<IRE domain>* is the FQDN of the primary server in the IRE. Use the same hostname that you used for the target primary server when you configured the SLPs in the production environment.
- *<IRE username>* is the username of a NetBackup user with permission to list SLPs and storage units in the IRE. For Windows users, enter the username in the format ***<domain name>\<username>***. For other users, enter the username only.

For example:

```
production_primary_server=examplePrimary.domain.com
production_primary_server_username=appadmin
ire_primary_server=exampleIREPrimary.domain.com
ire_primary_server_username=appadmin
```

The following is an example output of the command:

```
EveryDayAtNoon:
SLPs: SLP1
Sunday start: 12:00:00 duration: 00:59:59
Monday start: 12:00:00 duration: 00:59:59
Tuesday start: 12:00:00 duration: 00:59:59
Wednesday start: 12:00:00 duration: 00:59:59
Thursday start: 12:00:00 duration: 00:59:59
Friday start: 12:00:00 duration: 00:59:59
Saturday start: 12:00:00 duration: 00:59:59

WeeklyWindow:
SLPs: SLP2
Sunday start: 10:00:00 duration: 01:59:59
Monday NONE
Tuesday NONE
Wednesday NONE
Thursday NONE
Friday NONE
Saturday start: 10:00:00 duration: 01:59:59
```

This example shows two SLP windows:

- A daily window for one hour starting at noon.
- A weekly window for two hours starting at 10:00 A.M.

2 Based on the requirements for your environment, determine a schedule and take note of it. For an existing A.I.R. environment, the schedule must accommodate the SLP windows that you viewed in the previous step.

You can set a daily schedule that is open at the same time each day, or you can set a different schedule for each day of the week.

In the previous example, you can accommodate both SLP windows with either of the following:

- A daily schedule from 10:00 A.M. to 1:00 P.M.
- A schedule from 12:00 P.M. to 1:00 P.M. on Monday through Friday and a schedule from 10:00 A.M. to 1:00 P.M. on Saturday and Sunday

---

Note: If the production environment and the IRE are in different time zones, the schedule must begin only once per day in both time zones. For example, if one environment is in the Asia/Kolkata time zone and the other is in the America/New_York time zone, the following schedule in Kolkata is not supported: Tuesday start time 22:00:00 and Wednesday start time 03:00:00. When these start times get converted to the New York time zone, they become Tuesday start time 12:30:00 and Tuesday start time 17:30:00, which is not supported.

---

3 Run the following command to configure which subnets and IP addresses are allowed to access the WORM storage server:

```
setting ire-network-control allow-subnets subnets=<CIDR subnets
or IP addresses>
```

Where <*CIDR subnets or IP addresses*> is a comma-separated list of the allowed IP addresses and subnets, in CIDR notation.

For example:

```
setting ire-network-control allow-subnets
subnets=10.80.120.208,10.84.48.0/20
```

Note: The IRE primary server, the IRE media servers, and the DNS server for the IRE must be included in the allowed list. If all of these servers are in the same subnet, only the subnet is required to be in the allowed list. If you have a dual stack IPv4-IPv6 network, make sure that you add both the IPv4 and the IPv6 addresses to the allowed list.

---

4 Run the following command to set the daily air gap schedule:

```
setting ire-network-control set-schedule start_time=<time>
duration=<duration> [weekday=<0-6>]
```

Where [weekday=<*0-6*>] is an optional parameter to indicate the day if you need to set different schedules for different days. 0 is Sunday, 1 is Monday, etc.

For example:

```
setting ire-network-control set-schedule start_time=10:00:00
duration=03:00:00 weekday=0
```

5 Before you can send data between the production domain and the IRE storage server, you must add MSDP reverse connections and add the replication operation.

See "Configuring data transmission between a production environment and an IRE WORM storage server" on page 702.

Managing an isolated recovery environment on a WORM storage server

Once you have configured an isolated recovery environment (IRE) on a WORM storage server, you can manage it from the deduplication shell as the **msdpadm** user. Use the following commands.

- To view the SLP windows from the primary server to the WORM server:

  ```
  setting ire-network-control show-slp-windows
  production_primary_server=<production domain>
  production_primary_server_username=<production username>
  ire_primary_server=<IRE domain> ire_primary_server_username=<IRE
  username> [production_use_apikey] [ire_use_apikey]
  ```

  Where:

  - *<production domain>* is the fully qualified domain name (FQDN) of the primary server in your production environment.
  - *<production username>* is the username of a NetBackup user with permission to list SLPs and SLP windows in the production environment. For Windows users, enter the username in the format ***<domain name>\<username>***. For other users, enter the username only.
  - *<IRE domain>* is the FQDN of the primary server in the IRE. Use the same hostname that you used for the target primary server when you configured the SLPs in the production environment.

- *<IRE username>* is the username of a NetBackup user with permission to list SLPs and storage units in the IRE. For Windows users, enter the username in the format ***<domain name>\<username>***. For other users, enter the username only.

For example:

```
production_primary_server=examplePrimary.domain.com
production_primary_server_username=appadmin
ire_primary_server=exampleIREPrimary.domain.com
ire_primary_server_username=appadmin
```

Note: The SLP replication window on the production domain must be configured to be open at the same time as the IRE schedule.

- To list the MSDP reverse connections:

```
setting ire-network-control list-reverse-connections
```

- To add an MSDP reverse connection:

```
setting ire-network-control add-reverse-connection
remote_storage_server=<production MSDP server>
[remote_primary_server=<production primary server>]
[local_storage_server=<IRE network interface>]
```

Where:

  - *<production MSDP server>* is the FQDN of the MSDP server in your production environment.
  - [remote_primary_server=*<production primary server>*] is an optional parameter for the FQDN of the primary server in your production environment. This parameter is required if the IRE domain uses an alternative name to access the production primary server. This scenario usually occurs if the production primary server runs on multiple networks with multiple hostnames.
  - [local_storage_server=*<IRE network interface>*] is an optional parameter for the hostname of the network interface to use for image replication on the IRE storage server. This parameter is required if the network interface for replication is different than the IRE storage server name.

- To verify that a reverse connection works:

```
setting ire-network-control validate-reverse-connection
remote_storage_server=<production MSDP server>
[remote_primary_server=<production primary server>]
[local_storage_server=<IRE network interface>]
```

- To remove an MSDP reverse connection:

```
setting ire-network-control remove-reverse-connection
remote_storage_server=<production MSDP server>
```

- To view the allowed IP addresses and subnets:
  ```
  setting ire-network-control show-allows
  ```
- To add IP addresses and subnets to the allowed list:
  ```
  setting ire-network-control allow-subnets subnets=<CIDR subnets
  or IP addresses>
  ```
  Where *<CIDR subnets or IP addresses>* is a comma-separated list of the allowed IP addresses and subnets, in CIDR notation.

  For example:
  ```
  setting ire-network-control allow-subnets
  subnets=10.80.120.208,10.84.48.0/20
  ```

  Note: The IRE primary server, the IRE media servers, and the DNS server for the IRE must be included in the allowed list. If all of these servers are in the same subnet, only the subnet is required to be in the allowed list. If you have a dual stack IPv4-IPv6 network, make sure that you add both the IPv4 and the IPv6 addresses to the allowed list.

- To remove the IP addresses and subnets from the allowed list:
  ```
  setting ire-network-control allow-subnets subnets=,
  ```
- To view the daily air gap schedule:
  ```
  setting ire-network-control show-schedule
  ```
- To change the air gap schedule:
  ```
  setting ire-network-control set-schedule start_time=<time>
  duration=<duration>
  ```
  For example:
  ```
  setting ire-network-control set-schedule start_time=10:00:00
  duration=03:00:00
  ```

  Note: If the production environment and the IRE are in different time zones, the schedule must begin only once per day in both time zones. For example, if one environment is in the Asia/Kolkata time zone and the other is in the America/New_York time zone, the following schedule in Kolkata is not supported: Tuesday start time 22:00:00 and Wednesday start time 03:00:00. When these start times get converted to the New York time zone, they become Tuesday start time 12:30:00 and Tuesday start time 17:30:00, which is not supported.

- To stop the air gap schedule:

  ```
  setting ire-network-control delete-schedule
  ```

- To view the current network status and check whether the external network is open or closed:

  ```
  setting ire-network-control external-network-status
  ```

- To manually open the external network:

  ```
  setting ire-network-control external-network-open
  ```

- To manually close the external network and resume the air gap schedule:

  ```
  setting ire-network-control resume-schedule
  ```

Configuring data transmission between a production environment and an IRE WORM storage server

Once the configuration of an isolated recovery environment (IRE) is completed, the production NetBackup hosts are no longer able to access the WORM storage server. You need to add MSDP reverse connections to allow data transmission between the production MSDP storage server and the IRE WORM storage server. Then you can add the replication operation.

To configure data transmission between a production environment and an IRE

1. Open an SSH session to the IRE WORM storage server. Run the following command to determine if the external network is open:

   ```
   setting ire-network-control external-network-status
   ```

   If it is not, run the following command:

   ```
   setting ire-network-control external-network-open
   ```

2. Depending on the type of certificate authority that you use for host communication, do one of the following:

   - If you use a NetBackup Certificate Authority, run the following commands to request the certificates from the production domain:

     ```
     setting certificate get-CA-certificate
     primary_server=<production primary server>
     setting certificate get-certificate primary_server=<production
     primary server> token=<token>
     ```

   - If you use an external certificate authority, run the following commands to enroll the certificates with the production domain:

     ```
     setting certificate enroll-external-certificates
     server=<production primary server>
     ```

3 Run the following command to add an MSDP reverse connection:

```
setting ire-network-control add-reverse-connection
remote_storage_server=<production MSDP server>
[remote_primary_server=<production primary server>]
[local_storage_server=<IRE network interface>]
```

Where:

- *<production MSDP server>* is the fully qualified domain name (FQDN) of the MSDP server in your production environment.
- [remote_primary_server=*<production primary server>*] is an optional parameter for the FQDN of the primary server in your production environment. This parameter is required if the IRE domain uses an alternative name to access the production primary server. This scenario usually occurs if the production primary server runs on multiple networks with multiple hostnames.
- [local_storage_server=*<IRE network interface>*] is an optional parameter for the hostname of the network interface to use for image replication on the IRE storage server. This parameter is required if the network interface for replication is different than the IRE storage server name.

4 If necessary, repeat the previous step to add additional MSDP reverse connections.

5 If Auto Image Replication (AIR) is not already configured on the production domain, run the following command to copy the IRE schedule to the production domain as a storage lifecycle policy (SLP) window:

```
setting ire-network-control sync-ire-window
production_primary_server=<production primary server>
production_primary_server_username=<production username>
[slp_window_name=<SLP window name>] [production_use_apikey]
```

Where:

- *<production primary server>* is the FQDN of the primary server in your production environment.
- *<production username>* is the username of a NetBackup user with permission to list SLPs and SLP windows in the production environment. For Windows users, enter the username in the format ***<domain name>\<username>***. For other users, enter the username only.
- [slp_window_name=*<SLP window name>*] is an optional parameter to give a name for the SLP window. If you do not provide this parameter, the name of the SLP window is `IRE_DEFAULT_WINDOW`.

6 If you do not have them already, create a source SLP on the production primary server and a target import SLP on the IRE primary server. See the section "Creating a storage lifecycle policy" in the *NetBackup Deduplication Guide* for details.

Note: You cannot add the replication operation from NetBackup when you create the SLPs. Continue to the next step to add the replication operation.

---

7 Run the following command to add the IRE WORM storage server as a replication target of the production NetBackup domain and to add the replication operation to the SLP:

```
setting ire-network-control add-replication-op
production_primary_server=<production primary server>
production_primary_server_username=<production username>
production_storage_server=<production storage server>
ire_primary_server_username=<IRE username>
source_slp_name=<production SLP name> target_import_slp_name=<IRE
SLP name> target_storage_server=<target storage server>
target_storage_server_username=<target storage server username>
production_storage_unit=<MSDP storage unit> [slp_window_name=<slp
window name>] [production_use_apikey] [ire_use_apikey]
```

Where:

- *<production primary server>* is the FQDN of the primary server in your production environment.
- *<production username>* is the username of a NetBackup user with permission to list SLPs and SLP windows in the production environment. For Windows users, enter the username in the format ***<domain name>\<username>***. For other users, enter the username only.
- *<production storage server>* is the FQDN of the production storage server in your production environment.
- *<IRE username>* is the username for an administrator on the IRE primary server. For Windows users, enter the username in the format ***<domain name>\<username>***. For other users, enter the username only.
- *<source SLP name>* is the SLP name from the production primary server to add the replication operation to.
- *<target SLP name>* is the import SLP name from the IRE primary server.
- *<target storage server>* is the FQDN of the target WORM storage server in your IRE environment.

- *<target storage server username>* is the username for the storage credentials on the target WORM storage server. This username is the same as the **Username for storage** that you entered when you created the instance.
- *<MSDP storage unit>* is the name of the MSDP storage unit that is the replication source in the source SLP.
- [slp_window_name=*<slp window name>*] is an optional parameter for the name of the SLP window that is synced with the IRE schedule. This parameter must match the SLP window name from the previous step, if applicable. If you do not provide this parameter, the default name is used.

8 If you opened the external network at the beginning of this procedure, run the following command to close it and resume the air gap schedule:

```
setting ire-network-control resume-schedule
```

Replicating the backup images from the IRE domain to the production domain

If the backup images at the production domain are lost for some reasons, you may need to replicate the backup images from the IRE domain back to the production domain. You can also replicate these backup images to a new domain.

To replicate the backup images from the IRE domain to the production domain

1 Ensure that NetBackup certificate authority (NBCA) or an external certificate authority (ECA) are configured on the IRE storage server.

- For BYO media server: See the step 1 of the *Configuring A.I.R. for replicating backup images from production environment to IRE BYO environment* topic.
  See "Configuring A.I.R. for replicating backup images from production environment to IRE BYO environment" on page 692.
- For WORM storage server, Flex Scale, and Access Appliance: See the step 2 of the *Configuring data transmission between a production environment and an IRE WORM storage server* topic.
  See "Configuring data transmission between a production environment and an IRE WORM storage server" on page 702.

2 On the IRE primary server, configure replication target using a command line.

- Create a configuration file for adding a replication target with the following information:

`V7.5 "operation" " " string`	The value must be `set-replication` to add a new replication target.
`V7.5 " rephostname" " " string`	Specifies the replication target host name.
`V7.5 "replogin" " " string`	Specifies the replication target storage server username.
`V7.5 "reppasswd" " " string`	Specifies the replication target storage server password.
`V7.5 "repsourcevolume" " " string`	Specifies the replication source volume name.
`V7.5 "reptargetvolume" " " string`	Specifies the replication target volume name.

- Run the following command to apply the configuration file to add replication target:

  ```
  /usr/openv/netbackup/bin/admincmd/nbdevconfig -setconfig
  -storage_server <IRE storage server name> -stype PureDisk
  -configlist <config file name>
  ```

- Run the following command to update the disk pool information on the primary server:

  ```
  /usr/openv/netbackup/bin/admincmd/nbdevconfig -updatedp -stype
  PureDisk -dp <IRE disk pool name>
  ```

Sample:

```
[admin@ire-primary ~]# cat add-replication.txt
V7.5 "operation" "set-replication" string
V7.5 "rephostname" "msdp-prod.example.com" string
V7.5 "replogin" "pduser" string
V7.5 "reppasswd" "pdpass" string
V7.5 "repsourcevolume" "PureDiskVolume" string
V7.5 "reptargetvolume" "PureDiskVolume" string
[admin@ire-primary ~]#
/usr/openv/netbackup/bin/admincmd/nbdevconfig -setconfig
-storage_server msdp-ire.example.com -stype PureDisk -configlist
 add-replication.txt
Storage server msdp-ire.example.com has been successfully updated
[admin@ire-primary ~]#
/usr/openv/netbackup/bin/admincmd/nbdevconfig -updatedp -stype
PureDisk -dp dp-ire
```

```
The following replication properties have been added to the disk
 pool : ReplicationSource
Replication sources and/or targets have changed for disk volumes:
 PureDiskVolume
Update of disk pool replication properties succeeded
```

3 Ensure that the production primary server has one import SLP that targets the relevant MSDP storage unit. Create it if it does not exist.

4 Run the following command on the IRE primary server to manually replicate the backup images to the production domain.

```
/usr/openv/netbackup/bin/admincmd/nbreplicate -backupid <backup
id> -cn <local copy number> -rcn <copy number plus 101> -slp_name
reverse-air -target_sts <production MSDP storage server name>
```

For example,

```
/usr/openv/netbackup/bin/admincmd/nbreplicate -backupid
client1_1234567890 -cn 1 -rcn 102 -slp_name reverse-air
-target_sts msdp-prod.example.com
```

Note: Before you start the replication, ensure that the backup image with the same backup ID does not exist in the production domain. If it exists, you must expire the image at the production domain first. Otherwise, the new replicated image will not get automatically imported by the import SLP at the production primary server.

Chapter 17

Using the NetBackup Deduplication Shell

This chapter includes the following topics:

- About the NetBackup Deduplication Shell
- Managing users from the deduplication shell
- Managing VLAN interfaces from the deduplication shell
- Managing the retention policy on a WORM storage server
- Managing images with a retention lock on a WORM storage server
- Auditing WORM retention changes
- Protecting the MSDP catalog from the deduplication shell
- About the external MSDP catalog backup
- Managing certificates from the deduplication shell
- Managing FIPS mode from the deduplication shell
- Managing post-quantum cryptography (PQC) mode from the deduplication shell
- Encrypting backups from the deduplication shell
- Tuning the MSDP configuration from the deduplication shell
- Setting the MSDP log level from the deduplication shell
- Managing NetBackup services from the deduplication shell
- Monitoring and troubleshooting NetBackup services from the deduplication shell

- Managing S3 service from the deduplication shell
- Multi-person authorization for deduplication shell commands
- Managing cloud LSU in Flex Scale and Cloud Scale
- Managing the NFS version 3 server services for the MSDP container
- Viewing the NetBackup RBAC roles assigned to the MSDP container

About the NetBackup Deduplication Shell

You can use the NetBackup Deduplication Shell to manage the settings for WORM and MSDP storage servers on the following products:

- Flex Appliance: Supported for WORM
- Access Appliance: Supported for WORM
- NetBackup Flex Scale: Supported for WORM and regular MSDP
- NetBackup on Azure Kubernetes Services (AKS): Supported for regular MSDP
- NetBackup on Amazon Elastic Kubernetes Service (EKS): Supported for regular MSDP

The interface provides tab-completed command options.

These are the main categories of commands:

- `dedupe`
  This command lets you manage the deduplication service.
- `retention`
  This command lets you manage image retention. This command is only available for WORM storage servers.
- `setting`
  This command lets you manage the deduplication and the system configuration settings.
- `support`
  This command lets you access and upload the relevant logs and configuration files for troubleshooting.
- `recovery`
  This command lets you run the MSDPCheck tool to check the data integrity and report possible corrupted images.

To access the deduplication shell, open an SSH session to the storage server. When you log in for the first time, use the following credentials:

- NetBackup Flex Scale: An appliance user with the appliance administrator role
- For Flex Appliance MSDP/WORM instances: Use the administrator password that was set for the `msdpadm` account in the Create Instance wizard. There is no default password, and no password change is required on first login.

Managing users from the deduplication shell

After you configure a WORM or an MSDP storage server, you can use the deduplication shell to add and manage additional users.

The following types of users are supported:

- Local users
  Local users for the storage server are managed from the deduplication shell for products other than NetBackup Flex Scale. For NetBackup Flex Scale, use the NetBackup Flex Scale management console. All users with the appliance administrator role have access to the deduplication shell.
  See "Adding and removing local users from the deduplication shell" on page 710.
- MSDP users
  See "Adding MSDP users from the deduplication shell" on page 711.
- Active Directory (AD) users for Universal Shares and Instant Access
  See "Connecting an Active Directory domain to a WORM or an MSDP storage server for Universal Shares and Instant Access" on page 713.

Adding and removing local users from the deduplication shell

Local users for the storage server are managed from the deduplication shell for products other than NetBackup Flex Scale. For NetBackup Flex Scale, use the NetBackup Flex Scale management console. All users with the appliance administrator role have access to the deduplication shell.

Use the following procedures to add or remove local users from the deduplication shell.

Adding local users

To add a local user

1. Open an SSH session to the server as the **msdpadm** user.
2. (Optional) If you want to use a random password for the new user, generate one with the following command:

   ```
   setting user random-password
   ```

3 Run the following command:

```
setting user add-user username=<username>
```

Where *<username>* is the username of the user that you want to add. When the prompt appears, enter the password for the user.

The password must have between 15 and 32 characters and must include at least one uppercase letter, one lowercase letter, one number, and one special character (_.+~@={}?!).

4 Run the following commands to view the new user:

- `setting user show-user username=<username>`
  This command shows the information about the new user.
- `setting user list-users`
  This command shows a list of all local users.

Removing local users

To remove a local user

1 Open an SSH session to the server as the **msdpadm** user.

2 Run the following command:

```
setting user delete-user username=<username>
```

Where *<username>* is the username of the user that you want to remove.

Note: The **msdpadm** user cannot be removed.

Adding MSDP users from the deduplication shell

NetBackup requires an MSDP user to connect to the deduplication storage. One MSDP user is required when you configure the storage server. If you use multiple NetBackup domains with the WORM instance, you need to add an additional MSDP user for each NetBackup domain after you create the instance.

Use the following procedure to add MSDP users from the deduplication shell.

To add an MSDP user

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. (Optional) If you want to use a random password for the new user, generate one with the following command:

   ```
   setting MSDP-user random-password
   ```

3. Run the following command. In NetBackup Flex Scale, run this command on the primary node.

   ```
   setting MSDP-user add-MSDP-user username=<username>
   password=<password>
   ```

   Where *<username>* is the username of the user that you want to add, and *<password>* is a password for that user.

   The username must have between 4 and 30 characters and can include letters and numbers.

   The password must be between 15 and 32 characters and must include at least one uppercase letter, one lowercase letter, one number, and one special character (_.+~={}?!).

4. Run the following commands to view the new user. In NetBackup Flex Scale, run this command on the primary node.

   - `setting MSDP-user verify-user username=<username>`
     This command verifies the username and the password for the new user.
   - `setting MSDP-user list`
     This command shows a list of all MSDP users.

Adding MSDP admin alias users from the deduplication shell

Adding a disk volume of an MSDP server to an MSDP volume group (MVG) may require creating an alias for the MSDP default user on the MSDP server. The MVG server communicates with the MSDP server using its own credentials. If these credentials do not match the credentials of the MSDP server that the disk volume resides on, an alias MSDP user must be created on the MSDP server. The alias user credentials must match those of the MVG server.

To add an MSDP alias user

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run the following command. In NetBackup Flex Scale, run this command on the primary node.

   ```
   setting MSDP-user add-MSDP-admin-alias username=<username>
   password=<password> owner=<owner>
   ```

   Where *<username>* and *<password>* are the username and password of the MSDP user on the MVG server, and <owner> is the username of the MSDP admin user on the MSDP server.

   This command is not supported in the Deduplication shell if the MSDP container is version 20.5 or earlier.
3. Run the following command to verify the alias user. Ensure that the list of users contains the new alias user and that the alias user has the same data selection ID as the MSDP server's admin user.

   ```
   setting MSDP-user list
   ```

Note: The MSDP container must be accessed with root user to run the operations directly in the container. See "Configuring the MSDP server to be used by an MVG server having different credentials" on page 248.

Connecting an Active Directory domain to a WORM or an MSDP storage server for Universal Shares and Instant Access

You can connect an Active Directory (AD) user domain to a WORM or an MSDP storage server for Universal Shares and Instant Access.

Note: The AD domain is only used for Universal Shares and Instant Access. AD users are not currently supported on the deduplication shell.

Use the following procedure to connect an AD user domain from the deduplication shell.

To connect an AD user domain

1. Verify that the storage server is on the same network as the AD domain. If it is not, edit the settings so that the server can reach the domain.
2. Open the following ports between the storage server and the remote host if they are not already open:

- 139
- 445

3. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
4. Run the following command:

   ```
   setting ActiveDirectory configure ad_server=<server name>
   domain=<domain name> domain_admin=<username>
   ```

   Where *<server name>* is the AD server name, *<domain name>* is the domain that you want to connect, and *<username>* is the username of an administrator user on that domain.
5. When the prompt appears, enter the password for the domain administrator user.

Disconnecting an Active Directory domain from the deduplication shell

Use the following procedure to disconnect an Active Directory (AD) user domain from the deduplication shell.

To disconnect an AD user domain

1. Open an SSH session to the storage server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run the following command:

   ```
   setting ActiveDirectory unconfigure ad_server=<server name>
   domain=<domain name> domain_admin=<username>
   ```

   Where *<server name>* is the AD server name, *<domain name>* is the domain that you want to disconnect, and *<username>* is the username of an administrator user on that domain.
3. When the prompt appears, enter the password for the domain administrator user.

Changing a user password from the deduplication shell

Use the following procedures to change the password of a local user or an MSDP user from the deduplication shell.

Note: Remote directory user passwords cannot be changed from the shell. They must be changed from the server on which they reside.

Changing a local user password

Use the following procedure to change the password of a local user or the default **msdpadm** user.

To change a local user password

1. Open an SSH session to the server as the user that you want to change the password for. You can also log in as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. (Optional) If you want to use a random password, generate one with the following command:

   ```
   setting user random-password
   ```

3. Run the following command:

   ```
   setting user change-password username=<username>
   ```

   Where *<username>* is the username of the user whose password you want to change.

4. Follow the prompt to change the password.

   The password must have between 15 and 32 characters and must include at least one uppercase letter, one lowercase letter, one number, and one special character (_.+~@={}?!).

5. (Optional) By default, passwords do not expire. To specify an expiration date for the password, run the following command:

   ```
   setting user set-password-exp-date username=<username>
   password_exp_date=<date>
   ```

   Where *<date>* is the expiration date in YYYY-MM-DD format.

   Once an expiration date has been set, you can view it with the following command:

   ```
   setting user show-password-exp-date username=<username>
   ```

Changing an MSDP user password

Use the following procedure to change the password of an MSDP user.

To change an MSDP user password

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. (Optional) If you want to use a random password, generate one with the following command:

   ```
   setting MSDP-user random-password
   ```

3. Run one of the following commands:

   - Reset the password:

     ```
     setting MSDP-user reset-password username=<username>
     ```

     Follow the prompt to enter the new password.

   - Validate and change the password on NetBackup Flex Scale:

     ```
     setting MSDP-user change-password username=<username>
     ```

     Enter the existing password and then a new password.

   The password must be between 15 and 32 characters and must include at least one uppercase letter, one lowercase letter, one number, and one special character (_.+~={}?!).

Managing VLAN interfaces from the deduplication shell

If you use multiple NetBackup domains with a WORM or an MSDP storage server, you can add additional VLAN interfaces for the server to connect with the other domains. Use the following procedures to manage the VLAN interfaces.

Adding a VLAN interface

To add a VLAN interface

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run the following command:

   ```
   setting MSDP-VLAN add interface=<VLAN IP address>
   ```

Removing a VLAN interface

To remove a VLAN interface

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run the following command:

```
setting MSDP-VLAN remove interface=<VLAN IP address>
```

Viewing the VLAN interfaces

To view the VLAN interfaces

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run the following command:

```
setting MSDP-VLAN list
```

Managing the retention policy on a WORM storage server

The WORM retention policy defines how long the saved data on a WORM storage server is protected with immutability and indelibility. The initial retention policy is determined when you configure the server. Use the following procedures to view or change the policy from the deduplication shell.

Viewing the retention policy

To view the retention policy

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run the following command:

```
setting WORM status
```

Changing the retention policy

Use the following procedure to change the retention policy. The new retention policy applies for future backups. It does not apply to backups that were taken before you made the change.

To change the retention policy

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run the following command and specify the minimum duration to keep the storage immutable and indelible:

   ```
   setting WORM set-min worm_min=<duration in seconds>
   ```

3. Run the following command and specify the maximum duration to keep the storage immutable and indelible:

   ```
   setting WORM set-max worm_max=<duration in seconds>
   ```

Note: This operation supports multi-person authorization. See "Multi-person authorization for deduplication shell commands" on page 757.

Managing images with a retention lock on a WORM storage server

The backup images on a WORM storage server have a retention lock based on the retention policy. The retention lock prevents the images from being modified or deleted. Use the following procedures to manage the backup images with a retention lock from the deduplication shell.

Note: You can also run the `catdbutil` command in the shell to manage the images. This command does not appear in the shell menu, but you can run it directly. However, the arguments for the command cannot include path separators (/). See "About the NetBackup command line options to configure immutable and indelible data" on page 225.

Viewing the backup images with a retention lock

To view the backup images with a retention lock

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run the following command:

   ```
   retention policy list
   ```

Disabling the retention lock on a backup image

You can disable the retention lock on a backup image if the appliance is in enterprise mode. You cannot disable the lock if the appliance is in compliance mode.

To disable the retention lock

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run one of the following commands:
   - To disable the retention lock on a single backup image:
     `retention policy disable backup_id="<ID>" copynumber=<number>`
   - To disable the retention lock on multiple backup images with the same copy number:
     `retention policy batch-disable backupids="<backupid1,backupid2,backupid3,...,backupidn>" copynumber=<number>`

   You can find the backup ID and the copy number in the output of the `retention policy list` command.

Note: This operation supports multi-person authorization. See "Multi-person authorization for deduplication shell commands" on page 757.

Auditing WORM retention changes

Use the following procedure to view a full history of the WORM configuration changes, including changes to the retention policy and to backup images.

Note: You can also run the `catdbutil` command in the shell to audit the retention changes. This command does not appear in the shell menu, but you can run it directly. However, the arguments for the command cannot include path separators (/). See "About the NetBackup command line options to configure immutable and indelible data" on page 225.

To audit retention changes

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run the following command:

   `retention policy audit`

Protecting the MSDP catalog from the deduplication shell

By default, WORM storage servers store a copy of the MSDP catalog in the directory `/mnt/msdp/vol0` in addition to the original copy that is available under the dedicated catalog volume (`/mnt/msdpcat`).

If MSDP is initially configured on the single volume (Flex Media, BYO, and Cloud Scale) and the first volume is lost, MSDP catalog copies are also lost and cannot be recovered. Now, when a new data volume is added, a catalog copy is added to the new data volume automatically if there is no dedicated catalog volume. Shadow copy of the catalog is available on the new data volume, and it can be recovered when the first volume is lost.

If you want extra protection for the catalog, you can configure additional copies. Use the following procedures to manage the MSDP catalog copies from the deduplication shell.

To view the catalog copies

1. Open an SSH session to the server.
2. Run the following command:

   ```
   cacontrol --catalog listshadowcopies
   ```

To configure an additional copy

1. Open an SSH session to the server.
2. Run the following command to determine which volumes exist in the `/mnt/msdp` directory:

   ```
   df -h
   ```

   Select one of the volumes other than vol0.

   Note: To configure an additional catalog copy, at least one volume other than vol0 must exist in the `/mnt/msdp` directory.

3. Run the following command:

   ```
   cacontrol --catalog addshadowcopy /mnt/msdp/<volume name>
   ```

   Where *<volume name>* is the volume that you chose in the previous step.

   For example:

   ```
   cacontrol --catalog addshadowcopy /mnt/msdp/vol1
   ```

About the external MSDP catalog backup

Use the external MSDP catalog backup utility to backup the MSDP catalog to an external storage server for the containerized MSDP (Flex WORM, Flex Scale, and Cloud Scale). To protect the MSDP catalog on other platforms, see the following topic.

See "About protecting the MSDP catalog" on page 201.

After the configuration, the utility takes backups of the MSDP catalog and imports them to NetBackup with an SLP. The SLP must be configured first to duplicate the imported MSDP catalog backup images to another storage server.

The following are the default catalog paths for the backup selection:

- `/database_path/databases/catalogshadow`
- `/storage_path/etc`
- `/database_path/databases/spa`
- `/storage_path/var`
- `/usr/openv/lib/ost-plugins/pd.conf`
- `/usr/openv/lib/ost-plugins/mtstrm.conf`
- `/database_path/databases/datacheck`

Configuring an external MSDP catalog backup from the deduplication shell

Use the `cacontrol` utility using the deduplication shell to configure and modify the external MSDP catalog backups. This utility is located at `/usr/openv/pdde/pdcr/bin`. To configure the external MSDP catalog backup, an MSDP user must be created with the app role. An existing user can be used, or new credentials can be provided, which can create a new MSDP user with the app role.

To set up the external MSDP catalog backup

1. On the NetBackup web UI, create an SLP with an import operation.

   Select the destination storage as the MSDP local LSU storage unit.

   Verify that the retention type is set to **Target retention**.

2. Add a child rule to the SLP.

   Select the operation to **Duplication** and set the destination storage to the desired external storage server to store the MSDP catalog backup.

   The duplication storage server cannot be the same as the MSDP storage server specified in step 1. Verify that the retention type is set to **Fixed** and set the retention period as desired.

3. Open an SSH session to the MSDP server.

4. Run the following command on the MSDP server to set up MSDP catalog backups.

   ```
   cacontrol --catalog setupexternalcopy <username> <password>
   <frequency in minutes> <slp_name>
   ```

   - `username and password`: These are the MSDP application user credentials. You may use an existing MSDP application user or provide new credentials. If new credentials are provided, the command automatically creates a new MSDP application user.
   - `frequency`: An interval to capture the MSDP catalog backup in minutes (1440 = daily, 10080 = weekly).
   - `slp_name`: The name of the SLP that was created in the step 1.
   - `username`/`password`: A username and password of the NetBackup user with the necessary rights or an administrator.

   It creates a configuration file in the MSDP data volume at `<STORAGE>/etc/puredisk/cat_backup.cfg` location. The configuration file cannot be changed manually. Log files and directories are created at `<STORAGE>/log/spad/` location.

To view the configuration

1. Open an SSH session to the MSDP server.

2. Run the following command to view the MSDP catalog backup configuration settings:

   ```
   cacontrol --catalog getexternalcopyconfig
   ```

To change the backup interval

1. Open an SSH session to the MSDP server.
2. Run the following command to change the backup interval:

   ```
   cacontrol --catalog editexternalcopyfrequency <frequency in
   minutes>
   ```

   The backup interval should only be changed after the external MSDP catalog backup is set up.

To change the import SLP name

1. On the NetBackup web UI, create an SLP with an import operation.

   Select the destination storage as the MSDP local LSU storage unit.

   Verify that the retention type is set to **Target retention**.
2. Add a child rule to the SLP.

   Select the operation to **Duplication** and set the destination storage to the desired external storage server to store the MSDP catalog backup.

   The duplication storage server cannot be the same as the MSDP storage server specified in step 1. Verify that the retention type is set to **Fixed** and set the retention period as desired.
3. Open an SSH session to the MSDP server.
4. Run the following command to change the SLP name that is used to import the MSDP catalog backup image to the NetBackup:

   ```
   cacontrol --catalog editexternalcopyslpname <slp_name>
   ```

   The SLP name should only be changed after the external MSDP catalog backup is set up.

Restoring from the external MSDP catalog backup

MSDP instance does not have NetBackup client, so the catalog image cannot be restored to a corrupted MSDP instance directly. Instead, the MSDP catalog backup must be restored to any NetBackup media server first. Then, it can be copied to the MSDP instance from the media server.

Troubleshooting the external MSDP catalog backup

If the SLP import operation is not configured correctly to match the MSDP storage server, backup jobs are imported with **SLP_No_Target_SLP** as the policy name and the duplication is not triggered.

To fix this issue, navigate to the SLP in the web UI and edit the SLP import operation. Change the destination storage to the local LSU MSDP storage unit.

Managing certificates from the deduplication shell

To authenticate NetBackup hosts, NetBackup uses security certificates that are issued by a Certificate Authority (CA). A NetBackup storage server can use either a NetBackup CA or an external CA. The CA is first configured when you configure the server. After configuration, you can use the deduplication shell to manage the CA certificates.

For more information on how NetBackup uses certificates, see the *NetBackup Security and Encryption Guide*.

Viewing the certificate details from the deduplication shell

To view the certificate details, log in to the deduplication shell as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.

Use the following commands to view the details of the current certificate configuration:

- `setting certificate list-certificates`
  Displays the details of all the host certificates that are available on the server
- `setting certificate list-CA-cert-details`
  Displays the details of all the CA certificates that are available on the server
- `setting certificate show-CA-cert-detail`
  Displays the NetBackup CA certificate details of the primary server that is currently in use
- `setting certificate show-external-CA-cert-detail`
  Displays the external CA certificate details of the primary server that is currently in use
- `setting certificate list-enrollment-status`
  Retrieves the enrollment status of the associated primary servers from the local certificate store
- `setting certificate show-CRL-check-level`
  Displays the revocation check level for the external certificates

Use the following commands to verify the status of the certificates:

- `setting certificate host-self-check`

Verifies whether the host certificate for the server is in the certificate revocation list (CRL)

- `setting certificate external-CA-health-check`
  Verifies the external certificates, the RSA keys, and the trust store

Importing certificates from the deduplication shell

Use the following procedures to import NetBackup or external certificates from the deduplication shell.

Importing a NetBackup certificate

To import a NetBackup certificate

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run one of the following commands:
   - To request the NetBackup CA certificate from the primary server:
     `setting certificate get-CA-certificate`
     By default, the command uses the first primary server entry in the NetBackup configuration file. You can specify an alternate primary server with the `primary_server` parameter. For example:
     `setting certificate get-CA-certificate primary_server=<alternate primary server hostname>`
   - To request a host certificate from the primary server:
     `setting certificate get-certificate [force=true]`
     Where [force=true] is an optional parameter that overwrites the existing certificate if it already exists.
     By default, the command uses the first primary server entry in the NetBackup configuration file. You can specify an alternate primary server with the `primary_server` parameter. For example:
     `setting certificate get-certificate primary_server=<alternate primary server hostname>`
     Depending on the primary server security level, the host may require an authorization or a reissue token. If the command prompts that a token is required for the request, enter the command again with the token for the host ID-based certificate. For example:
     `setting certificate get-certificate primary_server=<alternate primary server hostname> token=<certificate token> force=true`

Importing external certificates

To import external certificates

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run one of the following commands:
   - To download and install both the external CA certificate and the host certificate:

     `setting certificate install-external-certificates cacert=<trust store> cert=<host certificate> private_key=<key> [passphrase=<passphrase>] scp_host=<host> scp_port=<port>`

     Where:
     - <*trust store*> is the trust store in PEM format.
     - <*host certificate*> is the X.509 certificate of the host in PEM format.
     - <*key*> is the RSA private key in PEM format.
     - [passphrase=<*passphrase*>] is an optional parameter for the passphrase of the private key. This parameter is required if the key is encrypted.
     - <*host*> is the hostname of the host that stores the external certificates.
     - <*port*> is the port to connect to on the remote host.
   - To download and install the external CA certificate:

     `setting certificate get-external-CA-certificate cacert=<trust store> scp_host=<host> scp_port=<port>`

     Where:
     - <*trust store*> is the trust store in PEM format.
     - <*host*> is the hostname of the host that stores the external certificates.
     - <*port*> is the port to connect to on the remote host.
   - To download and install the external host certificate:

     `setting certificate get-external-certificates cert=<host certificate> private_key=<key> [passphrase=<passphrase>] scp_host=<host> scp_port=<port>`

     Where:
     - <*host certificate*> is the X.509 certificate of the host in PEM format.
     - <*key*> is the RSA private key in PEM format.

- [passphrase=<*passphrase*>] is an optional parameter for the passphrase of the private key. This parameter is required if the key is encrypted.
- <*host*> is the hostname of the host that stores the external certificates.
- <*port*> is the port to connect to on the remote host.

Note: If an external host certificate already exists on the server, it is overwritten.

3 (Optional) Run the following command to specify the revocation check level for the external certificates:

```
setting certificate set-CRL-check-level check_level=<DISABLE,
LEAF, or CHAIN>
```

The check levels are as follows:

- DISABLE: The revocation check is disabled. The revocation status of the certificate is not validated against the CRL during host communication.
- LEAF: The revocation status of the leaf certificate is validated against the CRL. LEAF is the default value.
- CHAIN: The revocation status of all certificates from the certificate chain is validated against the CRL.

Removing certificates from the deduplication shell

Use the following procedures to remove NetBackup or external certificates from the deduplication shell.

Warning: If you remove the existing certificates but have not installed new certificates, the WORM server can no longer communicate with the primary server. To switch from one type of certificate authority (CA) to the other, install the new NetBackup or external certificates before you remove the existing certificates.

To remove the NetBackup certificates

1 Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.

2 Run the following command:

```
setting certificate disable-CA
```

To remove the external certificates

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run the following command:

```
setting certificate remove-enrollment
```

Managing FIPS mode from the deduplication shell

You can use FIPS mode on a WORM or an MSDP storage server to conform to the Federal Information Process Standards (FIPS) 140-2. Use the following procedures to manage FIPS mode.

Viewing the FIPS mode

To check if FIPS mode is enabled or disabled

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run the following command:

```
setting FIPS status
```

Enabling FIPS mode

To enable FIPS mode

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run the following command:

```
setting FIPS enable
```

Disabling FIPS mode

To disable FIPS mode

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run the following command:

```
setting FIPS disable
```

Managing post-quantum cryptography (PQC) mode from the deduplication shell

You can enable post-quantum cryptography (PQC) mode on a WORM storage server to enable use of the PQC algorithm in TLS 1.3 Hybrid Key Exchange. PQC mode can be enabled only if FIPS mode is disabled.

To learn more about PQC support in NetBackup, refer to the *NetBackup Security and Encryption guide*.

Caution: NetBackup uses the OQS provider to support post-quantum cryptography (PQC) for secure communications. While the OQS provider is a significant step towards preparing for a quantum-safe future, it is currently widely used in experimental and research environments. Cohesity recommends that you thoroughly assess all associated risks and ensure that its usage aligns with your organization's security policies and compliance requirements before enabling PQC mode.

Use the following procedures to manage PQC mode.

Viewing the PQC mode status

To check if PQC mode is enabled or disabled

1. Open an SSH session on the server as the `msdpadm` user.
2. Run the following command:

```
setting PQC status
```

Enabling PQC mode

To enable PQC mode

1. Open an SSH session on the server as the `msdpadm` user.
2. Run the following command:

```
setting PQC enable
```

Disabling PQC mode

To disable PQC mode

1. Open an SSH session on the server as the `msdpadm` user.
2. Run the following command:

```
setting PQC disable
```

Encrypting backups from the deduplication shell

To encrypt backups on a WORM or an MSDP storage server, you can configure MSDP encryption with or without the Key Management Service (KMS).

Use the following procedures to configure encryption for your backups from the deduplication shell.

To configure MSDP encryption with KMS

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run the following command:

   `setting encryption enable-kms kms_server=<server> key_group=<key group>`

   Where <*server*> is the host name of the external KMS server and <*key group*> is the KMS server key group name.
3. To verify the KMS encryption status, run the `setting encryption kms-status` command.

To configure MSDP encryption without KMS

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run the following command:

   `setting encryption enable`
3. To verify the MSDP encryption status, run the `setting encryption status` command.

Converting legacy KMS to Key Encryption Key (KEK)

The `convert-legacy-kms` command migrates the legacy index-based KMS to KEK-based KMS. This migration unencrypts the SO records using the legacy KMS key and then re-encrypts the SO record using the active KEK.

To rotate keys for KEK encryption:

- Use the `rotate-kektag` command to create a new KEK and rotate SO records to the new KEK using the new three-tiered KMS system. In this system, KMS keys now encrypt KEKs which in turn encrypt SOs.
- The `rotate-kms-keys` command rotates the KMS keys under the new KMS system. KEKs, which are stored in the KMS proxy database, are unencrypted using the corresponding KMS key and then re-encrypted using the active KMS key.

Tuning the MSDP configuration from the deduplication shell

The default MSDP configuration should work for most installations. However, if you need to make adjustments, use the following commands to set or view the parameters.

Parameter	Description	Commands
`AllocationUnitSize`	The allocation unit size for the data on the server	To set the parameter: `setting set-MSDP-param allocation-unit-size value=<number of MiB>` To view the parameter: `setting get-MSDP-param allocation-unit-size`
`DataCheckDays`	The number of days to check the data for consistency	To set the parameter: `setting set-MSDP-param data-check-days value=<number of days>` To view the parameter: `setting get-MSDP-param data-check-days`
`LogRetention`	The length of time to keep logs	To set the parameter: `setting set-MSDP-param log-retention value=<number of days>` To view the parameter: `setting get-MSDP-param log-retention`
`MaxCacheSize`	The maximum size of the NetBackup Deduplication Engine (`spoold`) fingerprint cache	To set the parameter: `setting set-MSDP-param max-cache-size value=<number of GB>` To view the parameter: `setting get-MSDP-param max-cache-size`
`MaxRetryCount`	The maximum number of times to retry a failed transmission	To set the parameter: `setting set-MSDP-param max-retry-count value=<number of retry times>` To view the parameter: `setting get-MSDP-param max-retry-count`

Parameter	Description	Commands
`SpadLogging`	The log level for the NetBackup Deduplication Manager (`spad`)	To set the parameter: `setting set-MSDP-param spad-logging log_level=<value>` See "Setting the MSDP log level from the deduplication shell" on page 736. To view the parameter: `setting get-MSDP-param spad-logging`
`SpooldLogging`	The log level for the NetBackup Deduplication Engine (`spoold`)	To set the parameter: `setting set-MSDP-param spoold-logging log_level=<value>` See "Setting the MSDP log level from the deduplication shell" on page 736. To view the parameter: `setting get-MSDP-param spoold-logging`
`WriteThreadNum`	The number of threads for writing data to the data container in parallel	To set the parameter: `setting set-MSDP-param write-thread-num value=<number of threads>` To view the parameter: `setting get-MSDP-param write-thread-num`
`CloudDataCacheSize`	The default data cache size when the cloud LSU is added. Decrease this value if sufficient free space is not available.	To set the parameter: `setting set-MSDP-param cloud-data-cache-size value=<number>` To view the parameter: `setting get-MSDP-param cloud-data-cache-size`
`CloudMapCacheSize`	The default map cache size when the cloud LSU is added. Decrease this value if sufficient free space is not available.	To set the parameter: `setting set-MSDP-param cloud-map-cache-size value=<number>` To view the parameter: `setting get-MSDP-param cloud-map-cache-size`

Parameter	Description	Commands
`CloudMetaCacheSize`	The default meta cache size when the cloud LSU is added. Decrease this value if sufficient free space is not available.	To set the parameter: `setting set-MSDP-param cloud-meta-cache-size value=<number>` To view the parameter: `setting get-MSDP-param cloud-meta-cache-size`
`CloudUploadCacheSize`	The default upload cache size when the cloud LSU is added. The minimum value is 12 GiB.	To set the parameter: `setting set-MSDP-param cloud-upload-cache-size value=<number>` To view the parameter: `setting get-MSDP-param cloud-upload-cache-size`
`EnableLocalPredictive SamplingCache`	The parameter to enable or disable the local predictive sampling cache. Both `spoold` and `spad` have this parameter, and it should be synced between them.	To set the parameter: `setting set-MSDP-param enable-local-predictive-sampling-cache value=<true/false>` To view the parameter: `setting get-MSDP-param enable-local-predictive-sampling-cache`
`MaxPredictiveCacheSize`	The maximum size of the `spoold` predictive cache.	To set the parameter: `setting set-MSDP-param max-predictive-cache-size value=<number of bytes/%>` To view the parameter: `setting get-MSDP-param max-predictive-cache-size`

Parameter	Description	Commands
`MaxSamplingCacheSize`	The maximum size of the `spoold` sampling cache.	To set the parameter: `setting set-MSDP-param max-sampling-cache-size value=<number of bytes/%>` To view the parameter: `setting get-MSDP-param max-sampling-cache-size`
`UsableMemoryLimit`	The maximum usable memory size in `spoold`.	To set the parameter: `setting set-MSDP-param usable-memory-limit value=<number of bytes/%>` To view the parameter: `setting get-MSDP-param usable-memory-limit`
`MaxCacheSize(Cluster)`	The maximum size of the `spoold` fingerprint cache for all nodes in a cluster.	To set the parameter: `setting set-MSDP-param max-cache-size-cluster value=<number>` To view the parameter: `setting get-MSDP-param max-cache-size-cluster`
`MaxPredictiveCacheSize (Cluster)`	The maximum size of the `spoold` predictive cache for all nodes in a cluster.	To set the parameter: `setting set-MSDP-param max-predictive-cache-size-cluster value=<number of bytes>` To view the parameter: `setting get-MSDP-param max-predictive-cache-size-cluster`

Parameter	Description	Commands
`MaxSamplingCacheSize (Cluster)`	The maximum size of the `spoold` sampling cache for all nodes in a cluster.	To set the parameter: `setting set-MSDP-param max-sampling-cache-size-cluster value=<number of bytes>` To view the parameter: `setting get-MSDP-param max-sampling-cache-size-cluster`
`UsableMemoryLimit(Cluster)`	The maximum usable memory size in `spoold` for all nodes in a cluster.	To set the parameter: `setting set-MSDP-param usable-memory-limit-cluster value=<number>` To view the parameter: `setting get-MSDP-param usable-memory-limit-cluster`
`EnableLocalPredictiveSampling Cache(Cluster)`	The parameter to enable or disable the local predictive sampling cache for all nodes in cluster. Both `spoold` and `spad` have this parameter, and it should be synced between them.	To set the parameter: `setting set-MSDP-param enable-local-predictive-sampling-cache-cluster value=<true/false>` To view the parameter: `setting get-MSDP-param enable-local-predictive-sampling-cache-cluster`
`VpfsCloudFPIndexRemoval Threshold (cluster)`	The threshold to remove the cloud fingerprint index file for all nodes in the cluster. When the number of deleted data containers in a fingerprint index file is greater than the threshold, the fingerprint index file is removed.	To set the parameter: `setting set-MSDP-param vpfs-cloud-fpindex-removal-threshold value=<%>` To view the parameter: `setting get-MSDP-param vpfs-cloud-fpindex-removal-threshold`

Parameter	Description	Commands
`VpfsPCacheReloadThreshold (cluster)`	The threshold for `spoold` to reload fingerprint from fingerprint index file based on the fingerprint in `pcache` that is replaced. This applies to all nodes in the cluster.	To set the parameter: `setting set-MSDP-param vpfs-pcache-reload-threshold-cluster value=<%>` To view the parameter: `setting get-MSDP-param vpfs-pcache-reload-threshold-cluster`

Setting the MSDP log level from the deduplication shell

You can set the log level on a WORM or an MSDP storage server for the following MSDP services:

- The NetBackup Deduplication Manager (`spad`)
- The NetBackup Deduplication Engine (`spoold`)

To set the log level

1. Open an SSH session to the server.
2. Run one of the following commands:
   - `setting set-MSDP-param spad-logging log_level=<value>,[thread],[date],[timing],[silent]`
   - `setting set-MSDP-param spoold-logging log_level=<value>,[thread],[date],[timing],[silent]`

   Where:
   - *<value>* is one of the following:
     - minimal: enables the critical, error, authorization, and bug logs
     - short: enables all minimal logs and adds warning logs
     - long: enables all short logs and adds info logs
     - verbose: enables all long logs and adds notice logs
     - full: enables all verbose logs and adds trace messages (all available logs)
     - none: disables logging

- [thread] is an optional parameter to enable thread ID logging.
- [date] is an optional parameter to include the date at the beginning of each logged event.
- [timing] is an optional parameter to enable high-resolution timestamps.
- [silent] is an optional parameter to stop the logs from printing on the console or the screen.

For example:

```
setting set-MSDP-param spoold-logging log_level=full,thread
```

Managing NetBackup services from the deduplication shell

You can manage the following NetBackup services from the deduplication shell:

- The cyclic redundancy checking (CRC) service
  See "Managing the cyclic redundancy checking (CRC) service" on page 738.
- The content router queue processing (CRQP) service
  See "Managing the content router queue processing (CRQP) service" on page 739.
- The online checking service
  See "Managing the online checking service" on page 739.
- The compaction service
  See "Managing the compaction service" on page 740.
- The deduplication (MSDP) services
  See "Managing the deduplication (MSDP) services" on page 740.
- The Storage Platform Web Service (SPWS)
  See "Managing the Storage Platform Web Service (SPWS)" on page 742.
- The Veritas provisioning file system (VPFS)
  See "Managing the Veritas provisioning file system (VPFS) configuration parameters" on page 744.
  See "Managing the Veritas provisioning file system (VPFS) mounts" on page 745.
- The NGINX service
  See "Managing the NGINX service" on page 746.
- The SMB service
  See "Managing the SMB service" on page 747.

- The following deduplication utilities:
  - The deduplication manager utility (`cacontrol`)
    See "Oracle stream handler" on page 46.
  - The deduplication engine utility (`crcontrol`)
    See "Viewing storage usage within MSDP container files" on page 523.

Note: These commands do not appear in the shell menu, but you can run them directly. The arguments for these commands cannot include path separators (/).

Managing the cyclic redundancy checking (CRC) service

The cyclic redundancy checking (CRC) service is a data integrity check. Use the following procedures to manage the CRC service from the deduplication shell.

To view the status of the CRC service

1. Open an SSH session to the server.
2. Run one of the following commands:
   - To view the general status of the CRC service:
     ```
     dedupe CRC state
     ```
   - To view the status of the fix mode on the CRC service:
     ```
     dedupe CRC fixmode-state
     ```

To enable the CRC service or enable a different CRC mode

1. Open an SSH session to the server.
2. Run one of the following commands:
   - To enable the CRC service:
     ```
     dedupe CRC enable
     ```
   - To enable fast checking, which begins the check from container 64 and does not sleep between checking containers:
     ```
     dedupe CRC fast
     ```
     When the fast CRC ends, CRC behavior reverts to the behavior before fast checking was invoked.
   - To enable fix mode, which runs the check and attempts to fix any inconsistent metadata:
     ```
     dedupe CRC enable-fixmode
     ```

To disable the CRC service or fix mode

1. Open an SSH session to the server.
2. Run one of the following commands:
   - To disable the CRC service:
     ```
     dedupe CRC disable
     ```
   - To disable fix mode:
     ```
     dedupe CRC disable-fixmode
     ```

Managing the content router queue processing (CRQP) service

The content router queue processing (CRQP) service ensures that the internal databases are in sync with the storage. Use the following procedures to manage the CRQP service from the deduplication shell.

To view the status of the CRQP service

1. Open an SSH session to the server.
2. Run one of the following commands:
   - `dedupe CRQP status`
     This command shows the status from the last time that the CRQP service ran.
   - `dedupe CRQP info`
     This command shows information about the current activity of the CRQP service.

To start the CRQP service

1. Open an SSH session to the server.
2. Run the following command:
   ```
   dedupe CRQP start
   ```

Managing the online checking service

The online checking service is a data integrity check. Use the following procedures to manage the online checking service.

To start the online checking service

1. Open an SSH session to the server
2. Run the following command:
   ```
   dedupe online-check start
   ```

To view the status of the online checking service

1. Open an SSH session to the server.
2. Run the following command:

```
dedupe online-check status
```

To enable or disable the online checking service

1. Open an SSH session to the server.
2. Run one of the following commands:

```
dedupe online-check enable

dedupe online-check disable
```

Managing the compaction service

The compaction service removes unnecessary data from the MSDP catalog. Use the following procedures to manage the compaction service.

To view the status of the compaction service

1. Open an SSH session to the server.
2. Run the following command:

```
dedupe compaction state
```

To enable or disable the compaction service

1. Open an SSH session to the server.
2. Run one of the following commands:

```
dedupe compaction enable

dedupe compaction disable
```

To start the compaction service outside of the system schedule

1. Open an SSH session to the server.
2. Run the following command:

```
dedupe compaction start
```

Managing the deduplication (MSDP) services

The deduplication services operate the Media Server Deduplication Pool (MSDP) storage on the storage server. Use the following procedures to manage the MSDP services.

To view the status of the MSDP services

1. Open an SSH session to the server.
2. Run the following command:

```
dedupe MSDP status
```

To stop the MSDP services

1. Open an SSH session to the server.
2. Check the health monitor status using the following command:

```
setting health status
```

3. If the health monitor is enabled, stop the monitor using the following command:

```
setting health disable
```

4. After disabling the health monitor, use the following command to stop the MSDP services:

```
dedupe MSDP stop
```

To start the MSDP services

1. Open an SSH session to the server.
2. Start the MSDP services using the following command:

```
dedupe MSDP start
```

3. Start the health monitor using the following command:

```
setting health enable
```

Managing the MSDP services across the cluster

The deduplication services operate the Media Server Deduplication Pool (MSDP) storage on the storage server. Use the following procedures to manage the cluster MSDP services.

To view the status of the MSDP services

1. Open an SSH session to any MSDP engine in a cluster.
2. Run the following command to view the status of the MSDP services in a cluster.

```
dedupe MSDP-cluster status
```

The status of the `vpfsd` is also displayed.

To stop the MSDP services

1. Open an SSH session to any MSDP engine in a cluster.
2. Run the following command to stop the MSDP services in a cluster.

   ```
   dedupe MSDP-cluster stop
   ```

   The `vpfsd` service is also stopped.

To start the MSDP services

1. Open an SSH session to any MSDP engine in a cluster.
2. Run the following command to start the MSDP services in a cluster.

   ```
   dedupe MSDP-cluster start
   ```

   The `vpfsd` service is also started.

Managing the Storage Platform Web Service (SPWS)

The Storage Platform Web Service (SPWS) is a service for Instant Access and Universal Shares. Use the following procedures to manage the SPWS.

To view the status of the SPWS

1. Open an SSH session to the server.
2. Run the following command:

   ```
   dedupe spws status
   ```

To stop or start the SPWS

1. Open an SSH session to the server.
2. Run one of the following commands:

   ```
   dedupe SPWS stop
   ```

   ```
   dedupe SPWS start
   ```

To view an SPWS configuration parameter

1. Open an SSH session to the server.
2. Run the following command:

   ```
   setting spws-config get-spws-param spws_configkey=<parameter>
   ```

   Where `<parameter>` is the parameter that you want to view.

To change an SPWS configuration parameter

1. Open an SSH session to the server.
2. Run the following command:

   ```
   setting spws-config set-spws-param spws_configsection=<section>
   spws_configkey=<parameter> spws_configvalue=<value>
   ```

   Where `<section>` is the Samba configuration section, `<parameter>` is the parameter that you want to change, and `<value>` is the value that you want to change it to. For example:

   ```
   setting spws-config set-spws-param spws_configsection=Livemount
   spws_configkey=maxallowedusharebackupjobs spws_configvalue=5
   ```

Troubleshooting connection failures

If you experience persistent connection failures to the SPWS, use the following procedure to push the certificate from the SPWS to the primary NetBackup Web Service.

To push the certificate

1. Open an SSH session to the server.
2. Run the following command:

   ```
   dedupe spws push-spws-certificate
   ```

   Note: In a cluster environment, you must run this command from the catalog engine. It cannot be run from the other engines.

Managing Open Cloud Storage Daemon

Open Cloud Storage Daemon (ocsd) is a service responsible for interacting with the cloud.

To view the status of OCSD

1. Open an SSH session to the server.
2. Run the following command:

   ```
   dedupe ocsd status
   ```

To stop or start OCSD

1. Open an SSH session to the server.
2. Run one of the following commands:

```
dedupe ocsd stop
```

```
dedupe ocsd start
```

Managing the Veritas provisioning file system (VPFS) configuration parameters

The Veritas provisioning file system (VPFS) is a service for Instant Access and Universal Shares. The default VPFS configuration works for most environments. However, you can adjust the parameters if needed. Some of the parameters that may affect performance include:

- `numOfInstance`
  This parameter specifies the number of `vpfsd` instances. A universal share uses one `vpfsd` instance by default. In most cases, one instance is adequate. Increasing the number of `vpfsd` instances might improve universal share performance, although it also requires more CPU and memory. You can increase the number of `vpfsd` instances from 1 to up to 16 and distribute the shares cross all the `vpfsd` instances.
- `CloudCacheSize`
  This parameter specifies the local disk cache size. This option applies only to Universal Shares with object store and Instant Access with object store.

Use the following procedures to manage the VPFS configuration parameters.

To view a VPFS configuration parameter

1. Open an SSH session to the server.
2. Run the following command:

```
setting vpfs-config get-vpfs-param vpfs_configkey=<parameter>
```

Where *<parameter>* is the parameter that you want to view.

To change a VPFS configuration parameter

1. Open an SSH session to the server.
2. Run the following command:

   ```
   setting vpfs-config set-vpfs-param vpfs_configkey=<parameter>
   vpfs_configvalue=<value>
   ```

   Where *<parameter>* is the parameter that you want to change, and *<value>* is the value that you want to change it to. For example:

   ```
   setting vpfs-config set-vpfs-param vpfs_configkey=numOfInstance
   vpfs_configvalue=2
   ```

To temporarily change the log level of a VPFS share

1. Open an SSH session to the server.
2. Run the following command:

   ```
   setting vpfs-config set-log-level vpfs_loglevel=<level>
   vpfs_shareid=<share ID>
   ```

   Where *<level>* is `debug`, `information`, or `error`, and *<share ID>* is the ID of the share that you want to change the log level of. For example:

   ```
   setting vpfs-config set-log-level vpfs_loglevel=debug
   vpfs_shareid=my-db-share
   ```

Note: This command is meant for a short debugging session, and the change is not preserved if you restart the instance. To permanently change the log level, use the following command:

```
setting vpfs-config set-vpfs-param vpfs_configkey=logLevel
vpfs_configvalue=<level>
```

The default log level is `information`.

Managing the Veritas provisioning file system (VPFS) mounts

The Veritas provisioning file system (VPFS) is a service for Instant Access and Universal Shares. Use the following procedures to manage the VPFS mounts.

To view the status of the VPFS mounts

1. Open an SSH session to the server.
2. Run the following command:

   ```
   dedupe vpfs status
   ```

To stop or start the VPFS mounts

1. Open an SSH session to the server.
2. Run one of the following commands:

   `dedupe vpfs stop`

   `dedupe vpfs force-stop` (Use this command only if the `dedupe vpfs stop` command does not work or becomes stuck.)

   `dedupe vpfs start`

To view the list of VPFS shares

1. Open an SSH session to the server.
2. Run the following command:

   `dedupe vpfs list-shares`

Managing the NGINX service

The NGINX service is the gateway for the Storage Platform Web Service (SPWS). Use the following procedures to manage the NGINX service.

To view the status of the NGINX service

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run the following command:

   `setting nginx status`

To stop or start the NGINX service

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run one of the following commands:

   `setting nginx stop`

   `setting nginx start`

Configuring the NGINX certificate

The NGINX certificate lets NGINX communicate with the NetBackup primary server. Use the following procedure to manage the configuration if there are issues with the certificate.

To configure the NGINX certificate

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run the following command:

   ```
   setting nginx config-cert
   ```

3. You can view the details of the NGINX certificate with the following command:

   ```
   setting nginx show-cert
   ```

Managing the SMB service

The SMB service includes the SMB users for Instant Access and Universal Shares. Use the following procedures to manage the SMB service.

To view the status of the SMB service

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run the following command:

   ```
   setting smb status
   ```

To stop or start the SMB service

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. Run one of the following commands:

   ```
   setting smb stop
   ```

   ```
   setting smb start
   ```

To view an SMB configuration parameter

1. Open an SSH session to the server.
2. Run the following command:

   ```
   setting smb get-smb-param smb_configkey=<parameter>
   ```

   Where `<parameter>` is the parameter you want to view.

To change an SMB configuration parameter

1. Open an SSH session to the server.
2. Run the following command:

   ```
   setting smb set-smb-param smb_configsection=<section>
   smb_configkey=<parameter> smb_configvalue=<value>
   ```

   Where `<section>` is the Samba configuration section, `<parameter>` is the parameter that you want to change, and `<value>` is the value that you want to change it to. For example:

   ```
   setting smb set-smb-param smb_configsection="global"
   smb_configkey="server signing" smb_configvalue="mandatory"
   ```

Monitoring and troubleshooting NetBackup services from the deduplication shell

You can use the following commands to monitor and troubleshoot the NetBackup services on a WORM or an MSDP storage server.

- The `setting health` command
  This command manages the health monitor on the server, which monitors the application high availability status.
  See "Managing the health monitor" on page 749.
- The `support` command
  This command lets you access logs and configuration files for troubleshooting.
  See "Viewing information about the system" on page 749.
  See "Viewing the deduplication (MSDP) history or configuration files" on page 750.
  See "Viewing the log files" on page 752.
  See "Collecting and transferring troubleshooting files" on page 754.
- The `setting kernel` command
  This command lets you search for a keyword in the kernel parameters.
  See "Viewing information about the system" on page 749.
- The `crstats` and `dcscan` commands

Note: These commands do not appear in the shell menu, but you can run them directly. The arguments for these commands cannot include path separators (/).

See "About the tool updates for cloud support" on page 320.

- The `msdpimgutil` command
  This command lets you check deduplication pool encryption status or image encryption status on the storage server.
  See "Checking the image encryption status" on page 526.

Managing the health monitor

The health monitor is enabled by default to monitor the application high availability status. Use the following procedures to manage the health monitor from the deduplication shell.

To view the status of the health monitor

1. Open an SSH session to the server.
2. Run the following command:

```
setting health status
```

To enable or disable the health monitor

1. Open an SSH session to the server.
2. Run one of the following commands:

```
setting health enable
```

```
setting health disable
```

Viewing information about the system

Use the following commands to view information about the system from the deduplication shell.

To view information about the hardware:

- `support hardware cpumem` or `support process htop`: Displays the CPU and memory information

To view information about the software:

- `support software show-MSDP-version`: Displays the Media Server Deduplication Pool (MSDP) version
- `support software show-OS-version`: Displays the Linux operating system version

To view information about the system processes:

- `support process MSDP-process`: Displays the MSDP processes
- `support process pidstat`: Displays the operating system processes (PIDs)

To view information about the system performance:

- `support diskio iostat`: Displays the information about the disk I/O
- `support diskio vmstat`: Displays the information about the wait on the disk I/O
- `support diskio nmon`: Displays the information about the monitor system, which monitors the disk I/O, the network I/O, and the CPU usage.
- `support diskio disk-volume`: Displays the information about the disk volume
- `support process memory-usage`: Displays the free and the used memory
- `support process atop`: Displays detailed information about the operating system and process activity

To search the kernel parameters:

- `setting kernel search-param keyword=<keyword>`
  Where *<keyword>* is the word that you want to search for.

To view network settings:

- `setting network ifconfig`: Displays the internet configuration
- `setting network ping ip=<ip_address>`: Pings the target IP address
- `setting network route route=<route>`: Displays the network routing information
- `setting network netstat`: Displays the network status and the protocol statistics

Viewing the deduplication (MSDP) history or configuration files

The deduplication services operate the Media Server Deduplication Pool (MSDP) storage on the storage server. You can view the following files from the MSDP services:

- The MSDP history files
- The MSDP configuration files

Use the following procedures to view or search these files from the deduplication shell.

To view information about the files

1. Open an SSH session to the server.
2. Run one of the following commands:
   - `support MSDP-history ls [dir=<directory>]`

- `support MSDP-config ls [dir=<directory>]`

  Where [dir=<*directory*>] is an optional parameter to specify the directory that you want to view the files from. For example:

  ```
  support MSDP-config ls dir=config
  ```

To view a file

1. Open an SSH session to the server.
2. Do one of the following:
   - To view an entire file, run one of the following commands:
     - `support MSDP-history cat file=<file>`
     - `support MSDP-config cat file=<file>`

     Where <*file*> is the file name of the file that you want to view.
   - To view the last 10 lines of a file, run one of the following commands:
     - `support MSDP-history tail file=<file>`
     - `support MSDP-config tail file=<file>`

     Where <*file*> is the file name of the file that you want to view.

To search a file

1. Open an SSH session to the server.
2. Run one of the following commands:
   - `support MSDP-history grep file=<file> pattern=<keyword>`
   - `support MSDP-config grep file=<file> pattern=<keyword>`

   Where `<file>` is the file name of the file that you want to search and `<keyword>` is the naming pattern that you want to search for. For example:

   ```
   support MSDP-config grep file=spa.cfg pattern=address
   ```

Viewing process information in the pseudo-file system

Use the following procedures to view or search files from the /proc/ directory.

To view information about the files

1. Open an SSH session to the server.
2. Run the command:

   ```
   support proc ls [dir=<directory>]
   ```

   - Where `[dir=<directory>]` is an optional parameter to specify the directory that you want to view the files from. For example:

```
support proc ls dir=config
```

To view a file

1. Open an SSH session to the server.
2. Do one of the following:
   - To view an entire file, run the command:
     ```
     support proc cat file=<file>
     ```
     Where `<file>` is the file name of the file that you want to view.
   - To view the last 10 lines of a file, run the following command:
     ```
     support proc tail file=<file>
     ```
     Where `<file>` is the file name of the file that you want to view.

To search a file

1. Open an SSH session to the server.
2. Run the following command:
   ```
   support proc grep file=<file> pattern=<keyword>
   ```
   Where `<file>` is the file name of the file that you want to search and `<keyword>` is the naming pattern that you want to search for. For example:
   ```
   support proc grep file=spa.cfg pattern=address
   ```

Viewing the deduplication rate of a Veritas provisioning file service (VPFS) share

Use the following procedure to view the deduplication rate of a Veritas provisioning file service (VPFS) share

To view the deduplication rate

1. Open an SSH session to the server.
2. Run the following command:
   ```
   dedupe vpfs meta-dump vpfs_metakind=dedupe vpfs_shareid=<share ID>
   ```
   Where <*share ID*> is the ID of the share that you want to view the deduplication rate of.

Viewing the log files

The following logs are available on a WORM storage server:

- The Media Server Deduplication Pool (MSDP) logs

These files include the logs for the `spad`, `spoold`, `ocsd`, and `vpfsd` services.

- The system logs
  These files include the logs in the `/mnt/nblogs` directory, which includes the logs related to instance management and certificates.

Use the following procedures to view or search the log files from the deduplication shell.

To view information about the files

1. Open an SSH session to the server.
2. Run one of the following commands:
   - `support MSDP-log ls [dir=<directory>]`
   - `support syslogs ls [dir=<directory>]`

   Where [dir=*<directory>*] is an optional parameter to specify the directory that you want to view the files from. For example:

   ```
   support MSDP-log ls dir=spoold
   ```

To view a file

1. Open an SSH session to the server.
2. Do one of the following:
   - To view an entire file, run one of the following commands:
     - `support MSDP-log cat file=<file>`
     - `support syslogs cat file=<file>`

     Where *<file>* is the file name of the file that you want to view.
   - To view the last 10 lines of a file, run one of the following commands:
     - `support MSDP-log tail file=<file>`
     - `support syslogs tail file=<file>`

     Where *<file>* is the file name of the file that you want to view.

To search a file

1. Open an SSH session to the server.
2. Run one of the following commands:
   - `support MSDP-log grep file=<file> pattern=<keyword>`
   - `support syslogs grep file=<file> pattern=<keyword>`

   Where `<file>` is the file name of the file that you want to search and `<keyword>` is the naming pattern that you want to search for. For example:

```
support MSDP-log grep file=spad* pattern=sessionStartAgent
```

Collecting and transferring troubleshooting files

You can collect files from the following categories and transfer them to another host for easier viewing:

- The Media Server Deduplication Pool (MSDP) history files
- The MSDP configuration files
- The MSDP log files
- The system log files

Use the following procedure to collect and transfer these files from the deduplication shell:

To collect and transfer files

1. Open an SSH session to the server as the **msdpadm** user, or for NetBackup Flex Scale, as an appliance administrator.
2. If you plan to collect and transfer a large log file, you may need to increase the amount of time before the SSH connection times out. The default is 10 minutes. Use the following steps to increase the time:
   - Run the following command:
     `setting ssh set-ssh-timeout ssh_timeout=<number of seconds>`
   - Run the following command to verify the change:
     `setting ssh show-ssh-timeout`
   - Close the current SSH session and open a new one.
3. Run one of the following commands to collect files of interest from the desired category:
   - `support MSDP-history collect`
   - `support MSDP-config collect`
   - `support MSDP-log collect`
   - `support syslogs collect`

   You can also use the following optional parameters:
   - `pattern=<keyword>`
     This parameter searches for a keyword within the files.
   - `mmin="<minutes, +minutes, or -minutes>"`

This parameter specifies the timeframe to collect files from, in minutes. To collect the files from x minutes ago, enter `mmin="x"`. To collect the files from less than x minutes ago, enter `mmin="-x"`. To collect the files from more than x minutes ago, enter `mmin="+x"`.

- `mtime="<days, +days, or -days>"`
  This parameter specifies the timeframe to collect files from, in days. To collect the files from x days ago, enter `mtime="x"`. To collect the files from less than x days ago, enter `mtime="-x"`. To collect the files from more than x days ago, enter `mtime="+x"`.

For example:

```
support MSDP-log collect pattern=spoold* mmin="+2"
```

4. Run the `scp` command from any category to create a tarball of all previously collected files (from all categories) and transfer the tarball to the target host using the `scp` protocol. For example:

```
support MSDP-config scp scp_target=user@example.com:/tmp
```

5. If applicable, run the following command to set the SSH time-out back to the default:

```
setting ssh set-ssh-timeout ssh_timeout=600
```

Verify the change with the `setting ssh show-ssh-timeout` command.

Managing S3 service from the deduplication shell

After you configure an MSDP WORM storage server on the Flex appliance, you can use the deduplication shell to configure and manage the S3 service.

- Configure the S3 service.
  See "Configuring the S3 service" on page 756.
- Create or reset the root credentials.
  See "Creating or resetting root credentials" on page 756.
- Change S3 service certificates.
  See "Changing the S3 service certificates" on page 756.
- Manage the S3 service status.
  See "Managing the S3 service " on page 757.

Configuring the S3 service

To use S3 interface for MSDP, you can configure S3 service on an MSDP WORM storage server instance.

To configure the S3 service

1. Open an SSH session to the server.
2. Run the following command:

```
setting s3srv s3srv-config s3srv_ca_type=<ca type>
[s3srv_loglevel=<log level>] [s3srv_port=<s3 port>]
```

   - **s3srv_ca_type**: Certificate authority type. NBCA: 1, ECA: 2.
   - **s3srv_loglevel**: S3 server log level.
     None: 0
     Error: 1
     Warning: 2
     Info: 3 (default)
     Debug: 4
   - **s3srv_port**: S3 server port. Default port is 8443.

Creating or resetting root credentials

After S3 interface for MSDP is configured, you can create root user's credentials to manage S3 credentials.

To create or reset root credentials of the S3 service

1. Open an SSH session to the server.
2. Run the following command:

```
setting s3srv s3srv-reset-iam
```

Changing the S3 service certificates

S3 server HTTPS certificate must be renewed manually when it expires.

To change the S3 service certificates

1. Open an SSH session to the server.
2. Run the following command:

```
setting s3srv s3srv-change-ca s3srv_ca_type=<ca type>
```

   - **s3srv_ca_type**: Certificate authority type. NBCA: 1, ECA: 2

Managing the S3 service

Perform the following steps to manage the S3 service status.

To manage the S3 service

1. Open an SSH session to the server.
2. View the status of the S3 service.

   ```
   setting s3srv status
   ```

3. Stop the S3 service.

   ```
   setting s3srv stop
   ```

4. Start the S3 services.

   ```
   setting s3srv start
   ```

Changing S3 service log level

Perform the following steps to manage the S3 service log level.

To change the S3 service log level

1. Open an SSH session to the server.
2. Run the following command:

   ```
   setting s3srv s3srv-change-loglevel s3srv_loglevel=<level>
   ```

   ```
   s3srv_loglevel: None: 0, Error: 1, Warning: 2, Info: 3, Debug: 4
   ```

Multi-person authorization for deduplication shell commands

On NetBackup web UI, the security administrator can configure multi-person authorization for critical MSDP operations. For more information about multi-person authorization, see the *Configuring multi-person authorization* chapter of the *NetBackup Security and Encryption Guide*.

If multi-person authorization is configured for an operation, an approval is required from the approver to perform that operation. When the MSDP administrator runs an operation command using the deduplication shell, a multi-person authorization ticket is created for the review. Only after the ticket is approved by an approver, the MSDP administrator is allowed to perform the operation.

For example, MSDP WORM configuration change requires multi-person authorization. When the MSDP administrator runs the `setting WORM set-max`

command to change the maximum duration of the WORM retention lock, a ticket is created for the review.

```
[msdp shell] > setting WORM set-max worm_max=1900000
Multi-person authorization is enforced. This operation requires
authorization from the primary server: primary.example.com.
Create a ticket for approval (y/n)? y
Ticket comment: enlarge the WORM lock duration.
Ticket 7 is created. Please wait for an approval and run the command
 again.
Operation failed
```

If the ticket is reviewed and approved by an approver, the MSDP administrator is allowed to run the command and perform the WORM configuration change.

```
[msdp shell] > setting WORM set-max worm_max=1900000
Multi-person authorization is enforced. This operation requires
authorization from the primary server: primary.example.com.
Ticket 7 for this operation is approved.
Operation completed successfully
```

If the ticket is still pending for approval, and the MSDP administrator runs the command again, the deduplication shell asks the administrator to wait for an approval.

```
[msdp shell] > setting WORM set-max worm_max=1900000
Multi-person authorization is enforced. This operation requires
authorization from the primary server: primary.example.com.
The ticket 7 for this operation is still under review. Please wait
for an approval and run the command again.
Operation failed
```

Managing cloud LSU in Flex Scale and Cloud Scale

Use deduplication Shell to manage and configure the settings for cloud logical storage unit (LSU).

To manage cloud WORM LSUs in Flex Scale and Cloud Scale

1. Open an SSH session to any MSDP engine in a cluster.
2. List the cloud WORM LSUs.

   ```
   setting WORM-Cloud list-lsu
   ```

3. Reload the WORM configuration.

   ```
   setting WORM-Cloud reload-config lsu_name=<lsu_name>
   ```

For more information about cloud LSU, see the *MSDP cloud support* chapter.

Managing the NFS version 3 server services for the MSDP container

You can manage the NFS version 3 server services for the MSDP container in Flex Scale using the NetBackup deduplication shell.

NFS version 3 is disabled by default in NetBackup Flex Scale 3.5.

To manage the NFS version 3 server services for the MSDP container

1. Open an SSH session to the server as an appliance administrator for NetBackup Flex Scale.
2. View the NFS server versions supported by the current kernel configuration.

   ```
   setting NFS get-nfs-versions
   ```

3. Enable NFS version 3 server services for the MSDP container.

   ```
   setting NFS enable-nfsv3
   ```

4. Disable NFS version 3 server services for the MSDP container.

   ```
   setting NFS disable-nfsv3
   ```

Viewing the NetBackup RBAC roles assigned to the MSDP container

The MSDP container requires the **Any MSDP Container** role to successfully call NetBackup APIs. Sometimes, a user may miss adding the required `MSDP_SERVER` entry to the NetBackup primary server configuration, or the entry may have been removed from the configuration.

In such cases, it can be difficult to troubleshoot and resolve the issue quickly. A new restricted shell command lets you quickly determine if the correct NetBackup RBAC roles have been assigned to the MSDP container.

To view the NetBackup RBAC roles assigned to the MSDP container:

1. Run the following command:

   ```
   setting certificate show-nb-web-token-roles
   ```

2. Check the output for the text: **Any MSDP Container**.

   If this text does not appear, it indicates that the `MSDP_SERVER` entry is missing from the NetBackup primary server configuration and must be added.

Chapter 18

Troubleshooting

This chapter includes the following topics:

- About unified logging
- About legacy logging
- NetBackup MSDP log files
- Troubleshooting MSDP configuration issues
- Troubleshooting MSDP operational issues
- Viewing MSDP disk errors and events
- MSDP event codes and messages
- Unable to obtain the administrator password to use an AWS EC2 instance that has a Windows OS
- Trouble shooting multi-domain issues
- Troubleshooting the cloud compaction error messages
- Troubleshooting the msdpcmdrun issues

About unified logging

Unified logging creates log file names and messages in a format that is standardized across Cohesity products. Only the `vxlogview` command can assemble and display the log information correctly. Server processes and client processes use unified logging.

Log files for originator IDs are written to a subdirectory with the name specified in the log configuration file. All unified logs are written to subdirectories in the following directory:

Windows	*install_path*\NetBackup\logs
UNIX	/usr/openv/logs

Note: Only the following types of users can access the logs: root and service users in Linux systems, and users present in the administrators group of Windows systems.

You can access logging controls in **Logging** host properties. You can also manage unified logging with the following commands:

vxlogcfg	Modifies the unified logging configuration settings.
vxlogmgr	Manages the log files that the products that support unified logging generate.
vxlogview	Displays the logs that unified logging generates. See "Examples of using vxlogview to view unified logs" on page 764.

About using the vxlogview command to view unified logs

Only the vxlogview command can assemble and display the unified logging information correctly. The unified logging files are in binary format and some of the information is contained in an associated resource file. These logs are stored in the following directory. You can display vxlogview results faster by restricting the search to the files of a specific process.

UNIX	/usr/openv/logs
Windows	*install_path*\NetBackup\logs

Table 18-1 Fields in vxlogview query strings

Field name	Type	Description	Example
PRODID	Integer or string	Provide the product ID or the abbreviated name of product.	PRODID = 51216 PRODID = 'NBU'
ORGID	Integer or string	Provide the originator ID or the abbreviated name of the component.	ORGID = 116 ORGID = 'nbpem'
PID	Long Integer	Provide the process ID	PID = 1234567

Table 18-1 Fields in `vxlogview` query strings *(continued)*

Field name	Type	Description	Example
`TID`	Long Integer	Provide the thread ID	`TID = 2874950`
`STDATE`	Long Integer or string	Provide the start date in seconds or in the locale-specific short date and time format. For example, a locale can have the format 'mm/dd/yy hh:mm:ss AM/PM'	`STDATE = 98736352` `STDATE = '4/26/11 11:01:00 AM'`
`ENDATE`	Long Integer or string	Provide the end date in seconds or in the locale-specific short date and time format. For example, a locale can have the format 'mm/dd/yy hh:mm:ss AM/PM'	`ENDATE = 99736352` `ENDATE = '04/27/11 10:01:00 AM'`
`PREVTIME`	String	Provide the hours in 'hh:mm:ss' format. This field should be used only with operators =, <, >, >=, and <=	`PREVTIME = '2:34:00'`
`SEV`	Integer	Provide one of the following possible severity types: 0 = `INFO` 1 = `WARNING` 2 = `ERR` 3 = `CRIT` 4 = `EMERG`	`SEV = 0` `SEV = INFO`
`MSGTYPE`	Integer	Provide one of the following possible message types: 0 = `DEBUG` (debug messages) 1 = `DIAG` (diagnostic messages) 2 = `APP` (application messages) 3 = `CTX` (context messages) 4 = `AUDIT` (audit messages)	`MSGTYPE = 1` `MSGTYPE = DIAG`

Table 18-1 Fields in `vxlogview` query strings *(continued)*

Field name	Type	Description	Example
`CTX`	Integer or string	Provide the context token as string identifier or `'ALL'` to get all the context instances to be displayed. This field should be used only with the operators = and `!=`.	`CTX = 78` `CTX = 'ALL'`

Table 18-2 Examples of query strings with dates

Example	Description
`(PRODID == 51216) && ((PID == 178964)|| ((STDATE == '2/5/15 09:00:00 AM') && (ENDATE == '2/5/15 12:00:00 PM'))`	Retrieves the log file message for the NetBackup product ID 51216 between 9AM and 12PM on 2015-05-02.
`((prodid = 'NBU') && ((stdate >= '11/18/14 00:00:00 AM') && (endate <= '12/13/14 12:00:00 PM'))) || ((prodid = 'BENT') && ((stdate >= '12/12/14 00:00:00 AM') && (endate <= '12/25/14 12:00:00 PM')))`	Retrieves the log messages for the NetBackup product NBU between 2014-18-11 and 2014-13-12 and the log messages for the NetBackup product BENT between 2014-12-12 and 2014-25-12.
`(STDATE <= '04/05/15 0:0:0 AM')`	Retrieves the log messages that were logged on or before 2015-05-04 for all of the installed Cohesity products.

Examples of using vxlogview to view unified logs

The following examples demonstrate how to use the `vxlogview` command to view unified logs.

Note: Only the following types of users can access the logs: root and service users in Linux systems, and users present in the administrators group of Windows systems.

Table 18-3 Example uses of the vxlogview command

Item	Example
Display all the attributes of the log messages	`vxlogview -p 51216 -d all`
Display specific attributes of the log messages	Display the log messages for NetBackup (51216) that show only the date, time, message type, and message text: `vxlogview --prodid 51216 --display D,T,m,x`
Display the latest log messages	Display the log messages for originator 116 (`nbpem`) that were issued during the last 20 minutes. Note that you can specify `-o nbpem` instead of `-o 116`: `# vxlogview -o 116 -t 00:20:00`
Display the log messages from a specific time period	Display the log messages for `nbpem` that were issued during the specified time period: `# vxlogview -o nbpem -b "05/03/15 06:51:48 AM"` `-e "05/03/15 06:52:48 AM"`
Display results faster	You can use the `-i` option to specify an originator for a process: `# vxlogview -i nbpem` The `vxlogview -i` option searches only the log files that the specified process (`nbpem`) creates. By limiting the log files that it has to search, `vxlogview` returns a result faster. By comparison, the `vxlogview -o` option searches all unified log files for the messages that the specified process has logged. **Note:** If you use the `-i` option with a process that is not a service, `vxlogview` returns the message "No log files found." A process that is not a service has no originator ID in the file name. In this case, use the `-o` option instead of the `-i` option. The -i option displays entries for all OIDs that are part of that process including libraries (137, 156, 309, etc.).
Search for a job ID	You can search the logs for a particular job ID: `# vxlogview -i nbpem | grep "jobid=`*`job_ID`*`"` The `jobid=` search key should contain no spaces and must be lowercase. When searching for a job ID, you can use any `vxlogview` command option. This example uses the `-i` option with the name of the process (`nbpem`). The command returns only the log entries that contain the job ID. It misses related entries for the job that do not explicitly contain the jobid=*job_ID*.

About legacy logging

In NetBackup legacy debug logging, a process creates log files of debug activity in its own logging directory. By default, NetBackup creates only a subset of logging directories, in the following locations:

Windows	*install_path*\NetBackup\logs *install_path*\Volmgr\debug
UNIX	/usr/openv/netbackup/logs /usr/openv/volmgr/debug

To use legacy logging, a log file directory must exist for a process. If the directory is not created by default, you can use the `mklogdir` utility to create the directories. Or, you can manually create the directories. When logging is enabled for a process, a log file is created when the process begins. Each log file grows to a certain size before the NetBackup process closes it and creates a new log file.

Note: To apply the appropriate permissions on the legacy log directories, always use the `mklogdir` utility present in Windows and Linux to create the legacy log directories for each platform.

You can use the following utility to create all of the log directories:

- Windows: *install_path*\NetBackup\Logs\mklogdir.bat
- UNIX: /usr/openv/netbackup/logs/mklogdir

Follow these recommendations when you create and use legacy log folders:

- Do not use symbolic links or hard links inside legacy log folders.
- Sometimes if a process runs for a non-root or non-admin user, no logging that occurs in the legacy log folders. In that case use the `mklogdir` command to create a folder for the required user.
- To run a command line for a non-root or non-admin user (troubleshooting when the NetBackup services are not running), create user folders for the specific command line. Create the folders either with the `mklogdir` command or manually with the non-root or non-admin user privileges.

Creating NetBackup log file directories for MSDP

Before you configure your NetBackup feature, create the directories into which the NetBackup commands write log files. Create the directories on the primary server

and on each media server that you use for your feature. The log files reside in the following directories:

- UNIX: `/usr/openv/netbackup/logs/`
- Windows:*install_path*`\NetBackup\logs\`

More information about NetBackup logging is available in the *NetBackup Logging Reference Guide*, available through the following URL:

https://support.cohesity.com/s/article/article-100040135

To create log directories for NetBackup commands

- Depending on the operating system, run one of the following scripts:

  UNIX: `/usr/openv/netbackup/logs/mklogdir`

  Windows: *install_path*`\NetBackup\logs\mklogdir.bat`

To create the `tpconfig` command log directory

- Depending on the operating system, create the `debug` directory and the `tpcommand` directory (by default, the `debug` directory and the `tpcommand` directory do not exist). The pathnames of the directories are as follows:

  UNIX: `/usr/openv/volmgr/debug/tpcommand`

  Windows: *install_path*`\Veritas\Volmgr\debug\tpcommand`

NetBackup MSDP log files

The NetBackup deduplication components write information to various log files. Some NetBackup commands or processes write messages to their own log files. Other processes use Other processes use Veritas Unified Logging (VxUL) log files. VxUL uses a standardized name and file format for log files. An originator ID (OID) identifies the process that writes the log messages.

See “About legacy logging” on page 766.

See “About unified logging” on page 761.

In VxUL logs, the messages that begin with an `sts` prefix relate to the interaction with the deduplication plug-in. Most interaction occurs on the NetBackup media servers. To view and manage VxUL log files, you must use NetBackup log commands. For information about how to use and manage logs on NetBackup servers, see the NetBackup Logging Reference Guide. The guide is available through the following URL:

Most interaction occurs on the NetBackup media servers. Therefore, the log files on the media servers that you use for disk operations are of most interest.

Warning: The higher the log level, the greater the affect on NetBackup performance. Use a log level of 5 (the highest) only when directed to do so by a Cohesity representative. A log level of 5 is for troubleshooting only.

Specify the NetBackup log levels in the **Logging** host properties on the NetBackup primary server. The log levels for some processes specific to certain options are set in configuration files as described in Table 18-4.

Table 18-4 describes the log files for each component.

Table 18-4 Logs for NetBackup MSDP activity

Component	VxUL OID	Description
Backups and restores	117	The `nbjm` Job Manager.
Backups and restores	N/A	Messages appear in the log files for the following processes: ■ The `bpbrm` backup and restore manager. The following is the path to the log files: UNIX: `/usr/openv/netbackup/logs/bpbrm` Windows: *`install_path`*`\Veritas\NetBackup\logs\bpbrm` ■ The `bpdbm` database manager. The following is the path to the log files: UNIX: `/usr/openv/netbackup/logs/bpdbm` Windows: *`install_path`*`\Veritas\NetBackup\logs\bpdbm` ■ The `bptm` tape manager for I/O operations. The following is the path to the log files: UNIX: `/usr/openv/netbackup/logs/bptm` Windows: *`install_path`*`\Veritas\NetBackup\logs\bptm`
Catalog shadow copies	N/A	The MSDP catalog shadow copy process writes messages to the following log files and directories: UNIX: *`/storage_path`*`/log/spad/spad.log` *`/storage_path`*`/log/spad/sched_CatalogBackup.log` *`/storage_path`*`/log/spad/`*`client_name`*`/` Windows: *`storage_path`*`\log\spad\spad.log` *`storage_path`*`\log\spad\sched_CatalogBackup.log` *`storage_path`*`\log\spad\`*`client_name`*`\`

Table 18-4 Logs for NetBackup MSDP activity *(continued)*

Component	VxUL OID	Description
Client deduplication proxy plug-in	N/A	The client deduplication proxy plug-in on the media server runs under `bptm`, `bpstsinfo`, and `bpbrm` processes. Examine the log files for those processes for proxy plug-in activity. The strings `proxy` or `ProxyServer` embedded in the log messages identify proxy server activity. They write log files to the following directories: ■ For `bptm`: UNIX: `/usr/openv/netbackup/logs/bptm` Windows: *`install_path`*`\Veritas\NetBackup\logs\bptm` ■ For `bpstsinfo`: Windows: `/usr/openv/netbackup/logs/admin` UNIX: `/usr/openv/netbackup/logs/bpstsinfo` Windows: *`install_path`*`\Veritas\NetBackup\logs\admin` Windows: *`install_path`*`\Veritas\NetBackup\logs\stsinfo` ■ For `bpbrm`: UNIX: `/usr/openv/netbackup/logs/bpbrm` Windows: *`install_path`*`\Veritas\NetBackup\logs\bpbrm`
Client deduplication proxy server	N/A	The deduplication proxy server `nbostpxy` on the client writes messages to files in the following directory, as follows: UNIX: `/usr/openv/netbackup/logs/nbostpxy` Windows: *`install_path`*`\Veritas\NetBackup\logs\nbostpxy`.
Deduplication configuration script	N/A	The following is the path name of the log file for the deduplication configuration script: ■ UNIX: *`storage_path`*`/log/pdde-config.log` ■ Windows: *`storage_path`*`\log\pdde-config.log` NetBackup creates this log file during the configuration process. If your configuration succeeded, you do not need to examine the log file. The only reason to look at the log file is if the configuration failed. If the configuration process fails after it creates and populates the storage directory, this log file identifies when the configuration failed.

Table 18-4 Logs for NetBackup MSDP activity *(continued)*

Component	VxUL OID	Description
Deduplication plug-in	N/A	The `DEBUGLOG` entry and the `LOGLEVEL` in the `pd.conf` file determine the log location and level for the deduplication plug-in. The following are the default locations for log files: ■ UNIX: `/var/log/puredisk/pdplugin.log` ■ Windows: `C:\pdplugin.log` You can configure the location and name of the log file and the logging level. To do so, edit the `DEBUGLOG` entry and the `LOGLEVEL` entry in the `pd.conf` file. See "About the MSDP pd.conf configuration file" on page 178. See "Editing the MSDP pd.conf file" on page 179.
Device configuration and monitoring	111	The `nbemm` process.
Device configuration and monitoring	178	The Disk Service Manager process that runs in the Enterprise Media Manager (EMM) process.
Device configuration and monitoring	202	The storage server interface process that runs in the Remote Manager and Monitor Service. RMMS runs on media servers.
Device configuration and monitoring	230	The Remote Disk Service Manager interface (RDSM) that runs in the Remote Manager and Monitor Service. RMMS runs on media servers.
`drcontrol` utility	N/A	You must run the `drcontrol` utility on the MSDP storage server host. The command requires administrator privileges. The utility creates a log file and displays its pathname in the command output. The utility writes log files to the following directory, depending on the operating system: UNIX: `/[storage_path]/log/drcontrol/policy_admin` `/storage_path/log/drcontrol/dedupe_catalog_DR` Windows: `storage_path\log\drcontrol\policy_admin` `storage_path\log\drcontrol\dedupe_catalog_DR` See "About protecting the MSDP catalog" on page 201.

Table 18-4 Logs for NetBackup MSDP activity *(continued)*

Component	VxUL OID	Description
Installation	N/A	The NetBackup installation process writes information about the installation of the deduplication components to a log file in the following directory: ■ UNIX: `/var/log/puredisk` ■ Windows: `%ALLUSERSPROFILE%\Symantec\NetBackup\InstallLogs`
NetBackup Deduplication Engine	N/A	The NetBackup Deduplication Engine writes several log files, as follows: ■ Log files in the *storage_path*`/log/spoold` directory, as follows: ■ The `spoold.log` file is the main log file ■ The `storaged.log` file is for queue processing. ■ The `storaged_<dsid>.log` file is for cloud LSU queue processing. ■ A log file for each connection to the engine is stored in a directory in the storage path `spoold` directory. The following describes the pathname to a log file for a connection: *hostname/application/TaskName/MMDDYY*.log For example, the following is an example of a `crcontrol` connection log pathname on a Linux system: /*storage_path*`/log/spoold/server.example.com/crcontrol/Control/010112.log` Usually, the only reason to examine these connection log files is if a Cohesity support representative asks you to. ■ A VxUL log file for the events and errors that NetBackup receives from polling. The originator ID for the deduplication engine is 364.
NetBackup Deduplication Engine	364	The NetBackup Deduplication Engine that runs on the deduplication storage server.

Table 18-4 Logs for NetBackup MSDP activity *(continued)*

Component	VxUL OID	Description
NetBackup Deduplication Manager	N/A	The log files are in the `/storage_path/log/spad` directory, as follows: ■ `spad.log` ■ `sched_QueueProcess.log` ■ `SchedClass.log` ■ A log file for each connection to the manager is stored in a directory in the storage path `spad` directory. The following describes the pathname to a log file for a connection: *hostname/application/TaskName/MMDDYY*.log For example, the following is an example of a `bpstsinfo` connection log pathname on a Linux system: `/storage_path/log/spoold/server.example.com/bpstsinfo/spad/010112.log` Usually, the only reason to examine these connection log files is if a Cohesity support representative asks you to. You can set the log level and retention period in the **Change Storage Server** dialog box **Properties** tab. See "Changing MSDP storage server properties" on page 534.
Optimized duplication and replication	N/A	For optimized duplication and Auto Image Replication, The following are the log files that provide information: ■ The NetBackup `bptm` tape manager for I/O operations. The following is the path to the log files: UNIX: `/usr/openv/netbackup/logs/bptm` Windows: `install_path\Veritas\NetBackup\logs\bptm` ■ The following is the path name of MSDP replication log file: `/storage_path/log/spad/replication.log`
Resilient network connections	387	The Remote Network Transport Service (`nbrntd`) manages resilient network connection sockets. It runs on the primary server, on media servers, and on clients. Use the VxUL originator ID 387 to view information about the socket connections that NetBackup uses. **Note:** If multiple backup streams run concurrently, the Remote Network Transport Service writes a large amount of information to the log files. In such a scenario, Cohesity recommends that you set the logging level for OID 387 to 2 or less. To configure unified logs, see the NetBackup Logging Reference Guide.

Table 18-4 Logs for NetBackup MSDP activity *(continued)*

Component	VxUL OID	Description
Resilient network connections	N/A	The deduplication plug-in logs information about keeping the connection alive. For more information about the deduplication plug-in log file, see “Deduplication plug-in” in this table. The `pd.conf` file `FILE_KEEP_ALIVE_INTERVAL` parameter controls the connection keep alive interval. See “About the MSDP pd.conf configuration file” on page 178. See “Editing the MSDP pd.conf file” on page 179.
The CRC notifications	N/A	The debug logs for CRC health check are saved at the following location: `/<storage_path>/log/spad/sched_CRCHealthCheck.log`

Troubleshooting MSDP configuration issues

The following sections may help you troubleshoot configuration issues.

See “NetBackup MSDP log files” on page 767.

See “MSDP storage server configuration fails” on page 773.

See “MSDP database system error (220)” on page 774.

See “MSDP server not found error” on page 774.

See “License information failure during MSDP configuration” on page 775.

See “The disk pool tab does not display an MSDP volume” on page 775.

MSDP storage server configuration fails

If storage server configuration fails, first resolve the issue that the **Storage server options** UI reports. Then, delete the deduplication host configuration file before you try to configure the storage server again.

NetBackup cannot configure a storage server on a host on which a storage server already exists. One indicator of a configured storage server is the deduplication host configuration file. Therefore, it must be deleted before you try to configure a storage server after a failed attempt.

See “Deleting an MSDP host configuration file” on page 200.

MSDP database system error (220)

A database system error indicates that an error occurred in the storage initialization.

Error message `ioctl() error, Database system error (220)`

Example

```
RDSM has encountered an STS error:

  Failed to update storage server ssname, database
  system error
```

Diagnosis

The `PDDE_initConfig` script was invoked, but errors occurred during the storage initialization.

First, examine the deduplication configuration script log file for references to the server name.

See "NetBackup MSDP log files" on page 767.

Second, examine the `tpconfig` command log file errors about creating the credentials for the server name. The `tpconfig` command writes to the standard NetBackup administrative commands log directory.

MSDP server not found error

The following information may help you resolve a server not found error message that may occur during configuration.

Error message `Server not found, invalid command parameter`

Example

```
RDSM has encountered an issue with STS where
  the server was not found:  getStorageServerInfo

   Failed to create storage server ssname, invalid
   command parameter
```

Diagnosis

Possible root causes:

- When you configured the storage server, you selected a media server that runs an unsupported operating system. All media servers in your environment appear in the **Storage server options** UI; be sure to select only a media server that runs a supported operating system.
- If you used the `nbdevconfig` command to configure the storage server, you may have typed the host name incorrectly. Also, case matters for the storage server type, so ensure that you use **PureDisk** for the storage server type.

License information failure during MSDP configuration

A configuration error message about license information failure indicates that the NetBackup servers cannot communicate with each other.

If you cannot configure a deduplication storage server or load balancing servers, your network environment may not be configured for DNS reverse name lookup.

You can edit the hosts file on the media servers that you use for deduplication. Alternatively, you can configure NetBackup so that it does not use reverse name lookup.

To prohibit reverse host name lookup by using the NetBackup web UI

1. Open the NetBackup web UI.
2. On the left, click **Host > Host Properties**.
3. If necessary, click **Connect**. Then click **Edit primary server**.
4. Click **Network settings**.
5. Select one of the following options:
   - **Allowed**
   - **Restricted**
   - **Prohibited**

   For a description of these options, see the *NetBackup Administrator's Guide, Volume I*.

To prohibit reverse host name lookup by using the `bpsetconfig` command

- Enter the following command on each media server that you use for deduplication:

  ```
  echo REVERSE_NAME_LOOKUP = PROHIBITED | bpsetconfig -h host_name
  ```

  The `bpsetconfig` command resides in the following directories:

  UNIX: `/usr/openv/netbackup/bin/admincmd`

  Windows: *`install_path`*`\Veritas\NetBackup\bin\admincmd`

The disk pool tab does not display an MSDP volume

The **Disk Pool tab** does not display a disk volume for the deduplication storage server.

First, restart all of the NetBackup daemons or services. The step ensures that the NetBackup Deduplication Engine is up and ready to respond to requests.

Second, refresh the **NetBackup web UI**. This step clears cached information from the failed attempt to display the disk volume.

Troubleshooting MSDP operational issues

The following sections may help you troubleshoot operational issues.

See "Verify that the MSDP server has sufficient memory" on page 776.

See "MSDP backup or duplication job fails" on page 776.

See "MSDP client deduplication fails" on page 779.

See "MSDP volume state changes to DOWN when volume is unmounted" on page 779.

See "MSDP errors, delayed response, hangs" on page 780.

See "Cannot delete an MSDP disk pool" on page 781.

See "MSDP media open error (83)" on page 782.

See "MSDP media write error (84)" on page 784.

See "MSDP no images successfully processed (191)" on page 785.

See "MSDP storage full conditions" on page 786.

See "Troubleshooting MSDP catalog backup" on page 786.

Verify that the MSDP server has sufficient memory

Insufficient memory on the storage server can cause operation problems. If you have operation issues, you should verify that your storage server has sufficient memory.

See "About MSDP server requirements" on page 37.

If the NetBackup deduplication processes do no start on Red Hat Linux, configure shared memory to be at least 128 MB (`SHMMAX=128MB`).

MSDP backup or duplication job fails

The following subsections describe some potential failures for backup or deduplication jobs and how to resolve them.

- Disk volume is down
- Storage server is down or unavailable
- Backup job: System error occurred (174)

- Failure to open storage path or to prepare CRQP transaction

Disk volume is down

A message similar to the following appears in the job details:

```
Error 800: Disk Volume is Down
```

Examine the disk error logs to determine why the volume was marked DOWN.

If the storage server is busy with jobs, it may not respond to primary server disk polling requests in a timely manner. A busy load balancing server also may cause this error. Consequently, the query times out and the primary server marks the volume DOWN.

If the error occurs for an optimized duplication job: verify that source storage server is configured as a load balancing server for the target storage server. Also verify that the target storage server is configured as a load balancing server for the source storage server.

See "Viewing MSDP disk errors and events" on page 788.

Storage server is down or unavailable

Windows servers only.

A message similar to the following appears in the job details:

```
Error nbjm(pid=6384) NBU status: 2106, EMM status: Storage Server is
down or unavailable Disk storage server is down(2106)
```

The NetBackup Deduplication Manager (`spad.exe`) and the NetBackup Deduplication Engine (`spoold.exe`) have different shared memory configuration values. This problem can occur when you use a command to change the shared memory value of only one of these two components.

To resolve the issue, specify the following shared memory value in the configuration file:

```
SharedMemoryEnabled=1
```

Then, restart both components. Do not change the values of the other two shared memory parameters.

The `SharedMemoryEnabled` parameter is stored in the following file:

storage_path`\etc\puredisk\agent.cfg`

Backup job: System error occurred (174)

A message similar to the following appears in the job details:

```
media manager - system error occurred (174)
```

If the job details also include errors similar to the following, it indicates that an image clean-up job failed:

```
Critical bpdm (pid=610364) sts_delete_image
 failed: error 2060018 file not found
Critical bpdm (pid=610364) image delete
 failed: error 2060018: file not found
```

This error occurs if a deduplication backup job fails after the job writes part of the backup to the **Media Server Deduplication Pool**. NetBackup starts an image cleanup job, but that job fails because the data necessary to complete the image clean-up was not written to the **Media Server Deduplication Pool**.

Deduplication queue processing cleans up the image objects, so you do not need to take corrective action. However, examine the job logs and the deduplication logs to determine why the backup job failed.

See "About MSDP queue processing" on page 552.

See "NetBackup MSDP log files" on page 767.

Failure to open storage path or to prepare CRQP transaction

Error messages similar to the following appear in one of the NetBackup Deduplication Engine (`spoold`) log files.

```
RefDBEngine::write_prepare fail to open
/storage_path/databases/refdb/prepare/64.ref.prepare
```

```
RefDBManager::write_prepare fail to prepare CRQP transaction for
refdb 64
```

See "NetBackup MSDP log files" on page 767.

This error occurs if the `/storage_path/databases/refdb/prepare` directory is deleted.

To fix this problem, do one of the following:

- Create the missing directory manually.
- Restart the NetBackup Deduplication Engine (`spoold`). First ensure that no backups are running on the storage unit on that media server.

Note: `RefDBEngine` and `refdb` do not refer to nor are they related to the open source RefDB reference database and bibliography tool.

MSDP client deduplication fails

NetBackup client-side agents (including client deduplication) depend on reverse host name look up of NetBackup server names. Conversely, regular backups depend on forward host name resolution. Therefore, the backup of a client that deduplicates it's own data may fail, while a normal backup of the client may succeed.

If a client-side deduplication backup fails, verify that your Domain Name Server includes all permutations of the storage server name.

Also, Cohesity recommends that you use fully-qualified domain names for your NetBackup environment.

See “Use fully qualified domain names” on page 51.

MSDP volume state changes to DOWN when volume is unmounted

If a volume becomes unmounted, NetBackup changes the volume state to DOWN. NetBackup jobs that require that volume fail.

To determine the volume state

- Invoke the following command on the primary server or the media server that functions as the deduplication storage server:

  UNIX: `/usr/openv/netbackup/bin/admincmd/nbdevquery -listdv -stype PureDisk -U`

  Windows: *`install_path`*`\NetBackup\bin\admincmd\nbdevquery -listdv -stype PureDisk -U`

  The following example output shows that the `DiskPoolVolume` is UP:

```
Disk Pool Name      : PD_Disk_Pool
Disk Type           : PureDisk
Disk Volume Name    : PureDiskVolume
Disk Media ID       : @aaaab
Total Capacity (GB) : 49.98
Free Space (GB)     : 43.66
Use%                : 12
Status              : UP
Flag                : ReadOnWrite
Flag                : AdminUp
Flag                : InternalUp
Num Read Mounts     : 0
Num Write Mounts    : 1
Cur Read Streams    : 0
Cur Write Streams   : 0
Num Repl Sources    : 0
Num Repl Targets    : 0
WORM Lock Min Time  : 0
WORM Lock Max Time  : 0
```

To change the volume state to UP

1. Mount the file system

   After a brief period of time, the volume state changes to UP. No further action is required.

2. If the volume state does not change, change it manually.

   See "Changing the MSDP disk volume state" on page 550.

MSDP errors, delayed response, hangs

Insufficient memory or inadequate host capabilities may cause multiple errors, delayed response, and hangs.

See “About MSDP server requirements” on page 37.

For virtual machines, Cohesity recommends that you do the following:

- Set the memory size of each virtual machine to double the physical memory of the host.
- Set the minimum and the maximum values of each virtual machine to the same value (double the physical memory of the host). These memory settings prevent the virtual memory from becoming fragmented on the disk because it does not grow or shrink.

These recommendations may not be the best configuration for every virtual machine. However, Cohesity recommends that you try this solution first when troubleshooting performance issues.

Cannot delete an MSDP disk pool

If you cannot delete a disk pool that you believe contains no valid backup images, the following information may help you troubleshoot the problem.

- Expired fragments remain on MSDP disk
- Incomplete SLP duplication jobs

Expired fragments remain on MSDP disk

Under some circumstances, the fragments that compose an expired backup image may remain on disk even though the images have expired. For example, if the storage server crashes, normal clean-up processes may not run. In those circumstances, you cannot delete a disk pool because image fragment records still exist. The error message may be similar to the following:

`DSM has found that one or more volumes in the disk pool diskpoolname has image fragments.`

To delete the disk pool, you must first delete the image fragments. The `nbdelete` command deletes expired image fragments from disk volumes.

To delete the fragments of expired images

- Run the following command on the primary server:

UNIX: `/usr/openv/netbackup/bin/admincmd/nbdelete -allvolumes -force`

Windows: `install_path\NetBackup\bin\admincmd\nbdelete -allvolumes -force`

The `-allvolumes` option deletes expired image fragments from all volumes that contain them.

The `-force` option removes the database entries of the image fragments even if fragment deletion fails.

Incomplete SLP duplication jobs

Incomplete storage lifecycle policy duplication jobs may prevent disk pool deletion. You can determine if incomplete jobs exist and then cancel them.

To cancel storage lifecycle policy duplication jobs

1. Determine if incomplete SLP duplication jobs exist by running the following command on the primary server:

   UNIX: `/usr/openv/netbackup/bin/admincmd/nbstlutil stlilist -image_incomplete`

   Windows: *install_path*`\NetBackup\bin\admincmd\nbstlutil stlilist -image_incomplete`

2. Cancel the incomplete jobs by running the following command for each backup ID returned by the previous command (*xxxxx* represents the backup ID):

   UNIX: `/usr/openv/netbackup/bin/admincmd/nbstlutil cancel -backupid` *xxxxx*

   Windows: *install_path*`\NetBackup\bin\admincmd\nbstlutil cancel -backupid` *xxxxx*

MSDP media open error (83)

The `media open error (83)` message is a generic error for the duplication. The error appears in the **Activity monitor**.

Often, the NetBackup Deduplication Engine (`spoold`) or the NetBackup Deduplication Manager (`spad`) were too busy to respond to the deduplication process in a timely manner. External factors may cause the Deduplication Engine or the Deduplication Manager to be unresponsive. Were they temporarily busy (such as queue processing in progress)? Do too many jobs run concurrently?

Usually but not always the NetBackup `bpdm` log provides additional information about status 83.

The following subsections describe use cases that generated an error 83.

SQL Server client-side backups fail

Client-side backups of a SQL Server database may fail in the following circumstances:

- The **Both IPv4 and IPv6** option is enabled for the primary server, the media server that hosts the NetBackup Deduplication Engine, and the client. The **Both IPv4 and IPv6** option is configured in the **Network settings** host properties.
- The IPv6 network is configured as a preferred network for the primary server, the media server that hosts the NetBackup Deduplication Engine, and the client. The preferred network **Match (Above network will be preferred for communication)** property also is enabled. Preferred networks are configured in the **Preferred network** host properties.
- The IPv6 network is chosen for the backup.

Examine the `bpbrm` log file for an error similar to the following:

```
probe_ost_plugin: sts_get_server_prop_byname failed: error 2060057
```

If the error message appears, the NetBackup host name cache may not contain the correct host name mapping information. The cache may be out of sync if DNS changes in your network environment were not fully propagated throughout your environment. It takes some amount of time for DNS changes to propagate throughout a network environment.

To resolve the problem, do the following on the NetBackup primary server and on the MSDP storage server:

1. Stop the NetBackup services.
2. Run the following command:

   UNIX: `/usr/openv/netbackup/bin/bpclntcmd -clearhostcache`

   Windows: *`install_path`*`\NetBackup\bin\bpclntcmd.exe -clearhostcache`
3. Start the NetBackup services.

For more information about client deduplication logging, see the description of “Client deduplication proxy plug-in” in the “MSDP log files” topic.

See “NetBackup MSDP log files” on page 767.

Restore or duplication fails

The `media open error (83)` message appears in the **Activity Monitor**.

Table 18-5 describes other messages that may appear.

Table 18-5 Case sensitivity error messages

Operation	Activity monitor job details	Status in `bpdm` and `bptm` log files
Restore	`Image open failed: error 2060018: file not found`	`sts_open_image failed: error 2060018`
Duplication (MSDP source)	`Image open failed: error 2060018: file not found`	`sts_open_image failed: error 2060018`
Replication (MSDP source)	`get image properties failed: error 2060013: no more entries`	`rpl_add_image_set: rpl_get_image_info() failed, error 2060013`

The messages may indicate a client name case sensitivity issue in your MSDP environment.

MSDP media write error (84)

Table 18-6 describes solutions to the media write errors that may occur during **Media Server Deduplication Pool** backups, duplication, or replication.

Also see the following subsections for descriptions of more complicated solutions:

- Host name resolution problems

Table 18-6 Media write error causes

The NetBackup Deduplication Engine (`spoold`) was too busy to respond.	Examine the Disk Logs report for errors that include the name PureDisk. Examine the disk monitoring services log files for details from the deduplication plug-in.
Data removal is running.	Data cannot be backed up at the same time as it is removed. See "About MSDP queue processing" on page 552.
A user tampered with the storage.	Users must not add files to, change files on, delete files from, or change file permissions on the storage. If a file was added, remove it.

Table 18-6 Media write error causes *(continued)*

Storage capacity was increased.	If you grew the storage, you must restart the NetBackup services on the storage server so the new capacity is recognized.
The storage is full.	If possible, increase the storage capacity. See "About provisioning the storage for MSDP" on page 57.
The deduplication pool is down.	Change the state to up.
Firewall ports are not open.	Ensure that ports 10082 and 10102 are open in any firewalls between the deduplication hosts.

Host name resolution problems

Client-side deduplication can fail if the client cannot resolve the host name of the server. More specifically, the error can occur if the storage server was configured with a short name and the client tries to resolve a fully qualified domain name

To determine which name the client uses for the storage server, examine the deduplication host configuration file on the client.

See "About the MSDP host configuration file" on page 199.

To fix this problem, configure your network environment so that all permutations of the storage server name resolve.

Cohesity recommends that you use fully qualified domain names.

See "Use fully qualified domain names" on page 51.

MSDP no images successfully processed (191)

The `no images successfully processed (191)` message appears in the Activity Monitor.

Table 18-7 describes other messages that may appear.

Table 18-7 Case sensitivity error messages

Operation	Activity Monitor job details	Status in `bpdm` and `bptm` log files
Verify	`image open failed: error 2060018: file not found`	`sts_open_image failed: error 2060018`

The message may indicate a client name case sensitivity issue in your MSDP environment.

MSDP storage full conditions

Operating system tools such as the UNIX `df` command do not report deduplication disk usage accurately. The operating system commands may report that the storage is full when it is not. NetBackup tools let you monitor storage capacity and usage more accurately.

See "About MSDP storage capacity and usage reporting" on page 521.

See "About MSDP container files" on page 523.

See "Viewing storage usage within MSDP container files" on page 523.

Examining the disk log reports for threshold warnings can give you an idea of when a storage full condition may occur.

How NetBackup performs maintenance can affect when storage is freed up for use.

See "About MSDP queue processing" on page 552.

See "About the MSDP data removal process" on page 563.

Although not advised, you can reclaim free space manually.

See "Processing the MSDP transaction queue manually" on page 553.

Troubleshooting MSDP catalog backup

The following subsections provide information about MSDP catalog backup and recovery.

Catalog backup

Table 18-8 describes any error messages that may occur when you create or update a catalog backup policy. The messages are displayed in the shell window in which you ran the `drcontrol` utility. The utility also writes the messages to its log file.

Table 18-8 MSDP `drcontrol` codes and messages

Code or message	Description
1	Fatal error in an operating system or deduplication command that the `drcontrol` utility calls.
110	The command cannot find the necessary NetBackup configuration information.
140	The user who invoked the command does not have administrator privileges.

Table 18-8 MSDP `drcontrol` codes and messages *(continued)*

Code or message	Description
144	A command option or argument is required.
226	The policy name that you specified already exists.
227	This error code is passed from the NetBackup `bplist` command. The MSDP catalog backup policy you specified does not exist or no backups exist for the given policy name.
255	Fatal error in the `drcontrol` utility.

For more information about status codes and error messages, see the following:

- The Troubleshooter in the NetBackup web UI
- *NetBackup Status Codes Reference Guide*

Catalog recovery from a shadow copy

If NetBackup detects corruption in the MSDP catalog, the Deduplication Manager recovers the catalog automatically from the most recent shadow copy. That recovery process also plays a transaction log so that the recovered MSDP catalog is current.

Although the shadow copy recovery process is automatic, a recovery procedure is available if you need to recover from a shadow copy manually.

Storage Platform Web Service (spws) does not start

Storage Platform Web Service (`spws`) does not start when you run `bp.start_all`.

Workaround:

If `spws` does not start when you run `bp.start_all`, run the following command to reconfigure `vpfs` and `spws`:

```
vpfs_config.sh --configure_byo
```

Disk volume API or command line option does not work

You have an MSDP storage server that has NetBackup version earlier than 8.3 and you have not enabled the encryption and KMS details. If you try to update the encryption and KMS details for a local volume using the new disk volume update API, the API operation succeeds. However, the actual values are not updated.

This issue occurs for both the API and command line option.

Viewing MSDP disk errors and events

You can view disk errors and events in several ways, as follows:

- The Disk Logs report.
- The NetBackup `bperror` command with the `-disk` option reports on disk errors. The command resides in the following directories:
  UNIX: `/usr/openv/netbackup/bin/admincmd`
  Windows: *`install_path`*`\Veritas\NetBackup\bin\admincmd`

MSDP event codes and messages

The following table shows the deduplication event codes and their messages. Event codes appear in the `bperror` command `-disk` output and in the disk reports in the NetBackup web UI.

Table 18-9 MSDP event codes and messages

Event #	Event Severity	NetBackup Severity	Message example
1000	2	Error	`Operation configload/reload failed on server PureDisk:server1.example.com on host server1.example.com.`
1001	2	Error	`Operation configload/reload failed on server PureDisk:server1.example.com on host server1.example.com.`
1002	4	Warning	`The open file limit exceeded in server PureDisk:server1.example.com on host server1.example.com. Will attempt to continue further.`
1003	2	Error	`A connection request was denied on the server PureDisk:server1.example.com on host server1.example.com.`
1004	1	Critical	`Network failure occurred in server PureDisk:server1.example.com on host server1.example.com.`
1008	2	Error	`Task Aborted; An unexpected error occurred during communication with remote system in server PureDisk:server1.example.com on host server1.example.com.`
1009	8	Authorization	`Authorization request from <IP> for user <USER> denied (<REASON>).`
1010	2	Error	`Task initialization on server PureDisk:server1.example.com on host server1.example.com got an unexpected error.`

Table 18-9 MSDP event codes and messages *(continued)*

Event #	Event Severity	NetBackup Severity	Message example
1011	16	Info	`Task ended on server PureDisk:server1.example.com on host server1.example.com.`
1013	1	Critical	`Task session start request on server PureDisk:server1.example.com on host server1.example.com got an unexpected error.`
1012	2	Error	`A request for agent task was denied on server PureDisk:server1.example.com on host server1.example.com.`
1014	1	Critical	`Task session start request on server PureDisk:server1.example.com on host server1.example.com got an unexpected error.`
1015	1	Critical	`Task creation failed, could not initialize task class on server PureDisk:server1.example.com on host server1.example.com.`
1017	1	Critical	`Service Cohesity DeduplicationEngine exit on server PureDisk:server1.example.com on host server1.example.com. Please check the server log for the probable cause of this error. The application has terminated.`
1018	16	Info	`Startup of Cohesity Deduplication Engine completed successfully on server1.example.com.`
1019	1	Critical	`Service Cohesity DeduplicationEngine restart on server PureDisk:server1.example.com on host server1.example.com. Please check the server log for the probable cause of this error. The application has restarted.`
1020	1	Critical	`Service Cohesity Deduplication Engine connection manager restart failed on server PureDisk:server1.example.com on host server1.example.com. Please check the server log for the probable cause of this error.The application has failed to restart.`
1028	1	Critical	`Service Cohesity DeduplicationEngine abort on server PureDisk:server1.example.com on host server1.example.com. Please check the server log for the probable cause of this error.The application has caught an unexpected signal.`

Table 18-9 MSDP event codes and messages *(continued)*

Event #	Event Severity	NetBackup Severity	Message example
1029	1	Critical	`Double backend initialization failure; Could not initialize storage backend or cache failure detected on host PureDisk:server1.example.com in server server1.example.com.`
1030	1	Critical	`Operation Storage Database Initialization failed on server PureDisk:server1.example.com on host server1.example.com.`
1031	1	Critical	`Operation Content router context initialization failed on server PureDisk:server1.example.com on host server1.example.com.`
1032	1	Critical	`Operation log path creation/print failed on server PureDisk:server1.example.com on host server1.example.com.`
1036	4	Warning	`Operation a transaction failed on server PureDisk:server1.example.com on host server1.example.com.`
1037	4	Warning	`Transaction failed on server PureDisk:server1.example.com on host server1.example.com. Transaction will be retried.`
1040	2	Error	`Operation Database recovery failed on server PureDisk:server1.example.com on host server1.example.com.`
1043	2	Error	`Operation Storage recovery failed on server PureDisk:server1.example.com on host server1.example.com.`
1044	multiple	multiple	`The usage of one or more system resources has exceeded a warning level. Operations will or could be suspended. Please take action immediately to remedy this situation.`
1057			`A data corruption has been detected.` The data consistency check detected a data loss or data corruption in the Media Server Deduplication Pool (MSDP) and reported the affected backups. The backup ID and policy name appear in the NetBackup Disk Logs report and the *storage_path*`/log/spoold/storaged.log` file on the storage server.
2000		Error	`Low space threshold exceeded on the partition containing the storage database on server PureDisk:server1.example.com on host server1.example.com.`

See "About MSDP storage capacity and usage reporting" on page 521.

See "Troubleshooting MSDP operational issues" on page 776.

See "MSDP events in NetBackup web UI" on page 529.

Unable to obtain the administrator password to use an AWS EC2 instance that has a Windows OS

This error occurs after the instance is launched from an AMI that is converted using automated disaster recovery.

The following error is displayed:

```
Password is not available. This instance was launched from a custom
AMI, or the default password has changed. A password cannot be
retrieved for this instance. If you have forgotten your password,
you can reset it using the Amazon EC2 configuration service. For more
information, see Passwords for a Windows Server Instance.
```

For more information, refer to the following articles:

- Amazon Elastic Compute Cloud Common Messages
- How to migrate your on-premises domain to AWS Managed Microsoft AD using ADMT

Trouble shooting multi-domain issues

The following sections may help you troubleshoot issues that involve multi-domain scenarios for NetBackup:

See "Unable to configure OpenStorage server from another domain" on page 791.

See "MSDP storage server is down when you configure an OpenStorage server" on page 792.

Unable to configure OpenStorage server from another domain

When you try to configure an OpenStorage server from another domain and the error `Login credentials verification failed for server xxxxxx` is displayed, try the following steps to find the root cause:

- Check if the user name and password are correct.
- Check if the NetBackup certificate is deployed to the media server that is used to configure the OpenStorage server. When certificate is not correctly deployed, the following error logs can be found in `pdplugin` log:

```
[ERROR] PDSTS: pd_register: PdvfsRegisterOST(egsuse1) failed
(30000:Unknown error 30000
[ERROR] PDSTS: get_agent_cfg_file_path_for_mount: pd_register()
failed for configuration file:</openv/lib/ost-plugins/egsuse1.cfg>
(2060401:UNKNOWN STS ERROR CODE)
```

For more information on using the `nbcertcmd` command to deploy NetBackup certificate for multi-domain, See "About MSDP multi-domain support" on page 215.

MSDP storage server is down when you configure an OpenStorage server

After configuring an OpenStorage server from another domain if the MSDP storage server is down or unresponsive, run the following steps find the root cause:

- Check if the same MSDP user is used by two or more NetBackup domains.
- Check if there is log entry in `spad.log` as follows:

  ```
  ERR [44] [140589294249728]: 25000: spaProcessing(), It's found that same
  msdp user "user1" is used by multiple NBU domains. This is wrong
  MultiDomain configuration which will cause potential data loss issue.
  Now other NBU domains cannot use msdp user "user1" to access MSDP
  services in this server.
  ```

  If there is an error log, the issue is that different NetBackup domains use the same MSDP user to access one MSDP storage server that is not supported by multi-domain.

- Backup or restore jobs fail with following error message in job details:

  ```
  Critical bptm (pid=140303) sts_open_server failed: error 2060405
  Wrong MSDP multi-domain configuration is detected. This NBU domain
  should use a different MSDP user to access the MSDP storage server.
  See NetBackup Deduplication Guide for details.
  ```

To resolve this issue

1. Create a new MSDP user in the MSDP storage server.

   See "About MSDP multi-domain support" on page 215.

   If the MSDP storage server is a NetBackup WORM Storage Server or NetBackup Flex Scale storage server, run following NetBackup Deduplication Shell command to create the MSDP user:

   ```
   setting MSDP-user add-MSDP-user username=<user_name> role=admin
   ```

2. If two NetBackup domains use the same MSDP user to access a MSDP server, update MSDP storage server credentials in the second NetBackup domain. The second NetBackup domain should use the new MSDP user to access the MSDP storage server. Run following command in every load balance server of the MSDP storage server in second NetBackup domain:

   ```
   tpconfig -update -stype PureDisk -storage_server
   <msdp_storage_server> -sts_user_id <user_name> -password
   <password>
   ```

3. Run following command to restart `spad` service in MSDP storage server.

   ```
   /usr/openv/pdde/pdconfigure/pdde spad stop
   ```

   ```
   /usr/openv/pdde/pdconfigure/pdde spad start
   ```

   If the MSDP storage server is a NetBackup WORM Storage Server or NetBackup Flex Scale storage server, run following NetBackup Deduplication Shell command to restart MSDP service:

   ```
   dedupe MSDP stop
   ```

   ```
   dedupe MSDP start
   ```

MSDP server is overloaded when it is used by multiple NetBackup domains

When the MSDP server is used by multiple NetBackup domains and the MSDP server has a high overload, run the following steps to check the workloads from the different domains:

1. Run the following command to get the current tasks status:

   For UNIX:

   ```
   /usr/openv/pdde/pdcr/bin/crcontrol --taskstat
   ```

   For Windows:

   <*install_path*>\Veritas\pdde\crcontrol.exe --taskstat

2. Check the client column for the list of clients that belong to the NetBackup domain, identify the work load of the clients from one domain, and then work load of one domain.
3. Run the `bpplclients` command on one NetBackup domain to list all clients of that domain.

Troubleshooting the cloud compaction error messages

Troubleshoot the following error messages for cloud compaction.

- **spoold** cannot start with the following error message:
  `Failed to recover from single container compaction.`
  spoold can not get anything from the cloud. Something may be wrong with OCSD.
  Ensure that the connection to cloud is working.
- Cloud compaction does not work with the following error message:
  `ingleDCCompactRecover: Call dcOutPlaceUpdateReplay_Cloud failed (...)`
  The container in cloud journal folder cannot be downloaded from `/data/compact_journal/*`.
  Ensure that the data container object in cloud journal folder can be downloaded.

Troubleshooting the msdpcmdrun issues

The following table lists the `msdpcmdrun` issues and how to troubleshoot those.

Table 18-10 The msdpcmdrun issues

Issue	Resolution
The following error message appears: "The operating system user must be part of the deduplication administrator group to run `msdpcmdrun` command."	The user attempting to run the `msdpcmdrun` command is not part of the deduplication administrator group. The administrator must add the user to the **pdadmin** group. See "Configuring a non-root user to run MSDP commands through msdpcmdrun " on page 230.

Table 18-10 The msdpcmdrun issues *(continued)*

Issue	Resolution
The following error message appears: "`msdpcmdrun` failed to connect to the admin service."	The `msdpcmdrun` is not able to communicate to the PureDisk Privileged Access Service (PDPAS) service. It may occur due to following reasons: ■ The PDPAS service is not running. ■ A new user is added to the **pdadmin** group but cannot access the PDPAS service. To resolve this issue: **1** Start the PDPAS service if it is not running. `/usr/openv/pdde/pdconfigure/pdde pdpas start` **2** If the PDPAS service is already running, restart it. `/usr/openv/pdde/pdconfigure/pdde pdpas stop` `ps -ef | grep pdpas` `/usr/openv/pdde/pdconfigure/pdde pdpas start`
The PDPAS service fails to stop using `bp.kill_all` command.	Sometimes, when attempting to stop the PDPAS service, the `bp.kill_all` command may fail, leaving the service running. To resolve this issue: **1** Stop the PDPAS service using the following command: `/usr/openv/pdde/pdconfigure/pdde pdpas stop` **2** If the process persists, locate the PDPAS process ID (PID) and kill it manually. ■ Identify the PID of the running PDPAS service: `ps -ef | grep pdpas` ■ Force kill the process: `kill -9 <PDPAS_PID>` ■ Verify that the process is no longer running: `ps -ef | grep pdpas`

Table 18-10 The msdpcmdrun issues *(continued)*

Issue	Resolution
After the PDPAS service is stopped, it starts running again.	A cron job runs every 5 minutes to monitor PDPAS. If it is not running, the PDPAS service is started automatically. If you want to stop the auto-start of the PDPAS service, you can disable the cron utility or remove the cron file, which monitors the PDPAS service. ■ To disable the cron utility, use operating system-specific command to stop the cron service. For example, on RHEL 8, use following command: `systemctl status crond.service` ■ To remove the cron file, run the following command: `rm /etc/cron.d/pdpas_monitor.cron`
One or both of the following error messages appear at the start of the output of the `msdpcmdrun` command: ■ "Failed to receive a response from the admin service." ■ "Error: -1: recvmsg failed: 88"	The `msdpcmdrun` can successfully communicate with the PDPAS service, but occasionally, race conditions in the operating system may cause error messages to appear at the start of the command output. To resolve this issue, restart the PDPAS service. `/usr/openv/pdde/pdconfigure/pdde pdpas stop` `ps -ef | grep pdpas` `/usr/openv/pdde/pdconfigure/pdde pdpas start`

Note: If you encounter any unexpected errors while running `msdpcmdrun`, restart the PDPAS service.

Appendix A

Migrating to MSDP storage

This appendix includes the following topics:

- Migrating from another storage type to MSDP

Migrating from another storage type to MSDP

To migrate from another NetBackup storage type to deduplication storage, Cohesity recommends that you age the backup images on the other storage until they expire. Cohesity recommends that you age the backup images if you migrate from disk storage or tape storage.

You should not use the same disk storage for NetBackup deduplication while you use it for other storage such as AdvancedDisk. Each type manages the storage differently and each requires exclusive use of the storage. Also, the NetBackup Deduplication Engine cannot read the backup images that another NetBackup storage type created. Therefore, you should age the data so it expires before you repurpose the storage hardware. Until that data expires, two storage destinations exist: the media server deduplication pool and the other storage. After the images on the other storage expire and are deleted, you can repurpose it for other storage needs.

Table A-1 Migrating to NetBackup MSDP

Step	Task	Procedure
Step 1	Configure NetBackup deduplication	See "Configuring media server deduplication in NetBackup" on page 63.

Table A-1 Migrating to NetBackup MSDP *(continued)*

Step	Task	Procedure
Step 2	Redirect your backup jobs	Redirect your backup jobs to the media server deduplication pool storage unit. To do so, change the backup policy storage destination to the storage unit for the deduplication pool. See the *NetBackup Administrator's Guide, Volume I*: https://support.cohesity.com/s/article/article-100040135
Step 3	Repurpose the storage	After all of the backup images that are associated with the storage expire, repurpose that storage. If it is disk storage, you cannot add it to an existing media server deduplication pool. You can use it as storage for another, new deduplication node.

Appendix B

Migrating from Cloud Catalyst to MSDP direct cloud tiering

This appendix includes the following topics:

- About migration from Cloud Catalyst to MSDP direct cloud tiering
- About Cloud Catalyst migration strategies
- About direct migration from Cloud Catalyst to MSDP direct cloud tiering
- About postmigration configuration and cleanup
- About the Cloud Catalyst migration -dryrun option
- About Cloud Catalyst migration cacontrol options
- Reverting back to Cloud Catalyst from a successful migration
- Reverting back to Cloud Catalyst from a failed migration

About migration from Cloud Catalyst to MSDP direct cloud tiering

Note: The procedures in this appendix must be run on MSDP servers running versions 10.0.0.1 through 10.2.0.1. The master server may be running a newer version, but the MSDP server on which the `nbdecommission -migrate_cloudcatalyst` command is run, must be running one of these versions.

NetBackup 8.3 and later releases include support for MSDP direct cloud tiering. This new technology is superior with improved performance, reliability, usability, and flexibility over the previous Cloud Catalyst product. You are encouraged to move to MSDP direct cloud tiering to take advantage of these improvements as well as future enhancements.

If you want to continue using Cloud Catalyst, you can do so on servers running NetBackup versions 8.1 through 8.3.0.2 because those versions are compatible with NetBackup 9.0 and later. Those older versions are supported as back-level servers for versions 9.0 and later NetBackup primary server installations. After you upgrade the NetBackup primary server to a version of 9.0 or later, you must use the command line to configure a Cloud Catalyst server. You cannot use the web UI with NetBackup 9.0 and later to configure Cloud Catalyst.

A `nbcheck` utility test has been added to the NetBackup install process to prevent Cloud Catalyst servers from being upgraded to version 9.0 and later. If Cloud Catalyst is detected on the server the install stops. The server remains unchanged, and continues to run the currently installed version of NetBackup after the upgrade is stopped.

About Cloud Catalyst migration strategies

Multiple strategies are available for migrating from Cloud Catalyst to MSDP direct cloud tiering. The best strategy for an installation depends on factors such as type of cloud storage (public versus private, standard versus cold storage class) and data retention requirements.

The following are four strategies for migrating from Cloud Catalyst to MSDP direct cloud tiering. Three of these strategies can be adopted with NetBackup 8.3 and later releases and the fourth, Direct Migration, is available in release 10.0 and later. All four strategies have advantages and disadvantages listed that you should review to help you make the best choice for your environment.

The four strategies for migrating from Cloud Catalyst to MSDP direct cloud tiering are as follows:

- Natural expiration strategy - Available in NetBackup release 8.3 and later.
- Image duplication strategy - Available in NetBackup release 8.3 and later.
- Combination strategy - Available in NetBackup release 8.3 and later.
- Direct migration strategy - Available in NetBackup release 10.0 and later.

Natural expiration strategy

This strategy works in any environment. To use this strategy, you must first configure a new NetBackup 8.3 or later MSDP direct cloud tier storage server. Or, add an

MSDP direct cloud tier disk pool and storage unit to an existing NetBackup 8.3 or later MSDP storage server (verify server capacity). Next, modify the storage lifecycle policies and backup policies to use the new MSDP direct cloud tier storage. Once all new duplication or backup jobs write to the new MSDP direct cloud tier storage, the images on the old Cloud Catalyst storage gradually expire. After all those images have expired, the Cloud Catalyst server can be retired or repurposed.

The advantages of the natural expiration strategy are as follows:

- Available with NetBackup version 8.3 and later. This strategy gives you improved performance, reliability, usability, and flexibility available in MSDP direct cloud tier. Can be used without upgrading to NetBackup 10.0.
- Can be implemented gradually using new MSDP Cloud storage servers while Cloud Catalyst storage servers continue to be used.
- Can be used for all environments including public cloud cold storage (for example: AWS Glacier or AWS Glacier Deep Archive).
- All new data is uploaded with the MSDP direct cloud tiering, which uses cloud storage more efficiently than Cloud Catalyst. The long-term total cloud storage usage and cost may be reduced.

The disadvantages of the natural expiration strategy are as follows:

- Until all the old Cloud Catalyst images have been expired and deleted, there is some duplication of data in cloud storage. This duplication can occur between the old Cloud Catalyst images and new MSDP direct cloud tier images. Additional storage costs could be incurred if you use a public cloud environment.
- Requires a separate server.
- Cloud Catalyst servers must be maintained until all uploaded images from those servers have expired or are otherwise no longer needed.

Image duplication strategy

This strategy works in most environments except those using public cloud cold storage (for example: AWS Glacier or AWS Glacier Deep Archive). To use this strategy, you must first configure a new NetBackup 8.3 or later MSDP direct cloud tier storage server. Or, add an MSDP direct cloud tier disk pool and storage unit to an existing NetBackup 8.3 or later MSDP storage server (verify server capacity). Next, modify the storage lifecycle policies and backup policies to use the new MSDP direct cloud tier storage. Once all new duplication or backup jobs write to the new MSDP direct cloud tier storage, existing images on the old Cloud Catalyst storage are moved. These images are moved to the new MSDP direct cloud tier storage using a manually initiated `bpduplicate` command. After all existing images have been moved from the old Cloud Catalyst storage to the new MSDP direct cloud tier storage, the Cloud Catalyst server can be retired or repurposed.

The advantages of the image duplication strategy are as follows:

- Available with NetBackup version 8.3 and later. This strategy gives you improved performance, reliability, usability, and flexibility available in MSDP direct cloud tier. Can be used without upgrading to NetBackup 10.0.
- Can be implemented gradually using new MSDP Cloud storage servers while Cloud Catalyst storage servers continue to be used.
- All new and all old Cloud Catalyst data is uploaded with MSDP direct cloud tiering, which uses cloud storage more efficiently than Cloud Catalyst. The long-term total cloud storage usage and cost may be reduced.

The disadvantages of the image duplication strategy are as follows:

- Public cloud cold storage environments (for example: AWS Glacier or AWS Glacier Deep Archive) support restore from the cloud but do not support duplication from the cloud, so this strategy cannot be used.
- If public cloud storage is used, potentially significant data egress charges are incurred when old Cloud Catalyst images are read to duplicate them to the new MSDP Cloud storage.
- Additional network traffic to and from the cloud occurs when the old Cloud Catalyst images are duplicated to the new MSDP direct cloud tier storage.
- Until all old Cloud Catalyst images have been moved to MSDP direct cloud tier storage, there is some duplication of data in cloud storage. This duplication can occur between the old Cloud Catalyst images and new MSDP direct cloud tier images. Additional costs could be incurred if you use a public cloud environment.
- Requires a separate server.
- Cloud Catalyst servers must be maintained until all uploaded images from those servers have been moved to the new MSDP direct cloud tier storage or are otherwise no longer needed.

Combination strategy

This strategy works in most environments except those using public cloud cold storage (example: AWS Glacier or AWS Glacier Deep Archive). This strategy is a combination of the previous two strategies. To use this strategy, you must first configure a new NetBackup 8.3 or later MSDP direct cloud tier storage server. Or, add an MSDP direct cloud tier disk pool and storage unit to an existing NetBackup 8.3 or later MSDP storage server (verify server capacity). Next, modify the storage lifecycle policies and backup policies to use the new MSDP direct cloud tier storage. Once all the new duplication or backup jobs write to the new MSDP direct cloud tier storage, the oldest images on the old Cloud Catalyst storage gradually expire. When the number of remaining unexpired images on the old Cloud Catalyst storage drops below a determined threshold, those remaining images are moved. These

images are moved to the new MSDP direct cloud tier storage using a manually initiated `bpduplicate` command. After all remaining images have been moved from the old Cloud Catalyst storage to the new MSDP direct cloud tier storage, the Cloud Catalyst server can be retired or repurposed.

The advantages of the combination strategy are as follows:

- Available with NetBackup version 8.3 and later. This strategy gives you improved performance, reliability, usability, and flexibility available in MSDP direct cloud tier. Can be used without upgrading to NetBackup 10.0.
- Can be implemented gradually using new MSDP direct cloud tier storage servers while Cloud Catalyst storage servers continue to be used.
- All new data and all old Cloud Catalyst data are uploaded with MSDP direct cloud tiering, which uses cloud storage more efficiently than Cloud Catalyst. The long-term total cloud storage usage and cost may be reduced.
- Enables retiring of the old Cloud Catalyst servers before all images on those servers have expired.

The disadvantages of the combination strategy are as follows:

- Public cloud cold storage environments (for example: AWS Glacier or AWS Glacier Deep Archive) support restore from the cloud but do not support duplication from the cloud, so this strategy cannot be used.
- If public cloud storage is used, potentially significant data egress charges are incurred. This issue can happen when old Cloud Catalyst images are read to duplicate them to the new MSDP direct cloud tier storage.
- Additional network traffic to and from the cloud occurs when the old Cloud Catalyst images are duplicated to the new MSDP direct cloud tier storage.
- Until all Cloud Catalyst images have expired or have been moved to MSDP direct cloud tier storage, there is some duplication of data in cloud storage. This duplication can occur between the old Cloud Catalyst images and new MSDP direct cloud tier images, so additional costs could be incurred if you use a public cloud environment.
- Requires a separate server.
- Cloud Catalyst servers must be maintained until all uploaded images from those servers have expired, have been moved to the new MSDP direct cloud tier, or are no longer needed.

Direct migration strategy

This strategy is available in NetBackup 10.0 and later releases and can work in any environment. To use this strategy, you must first configure a new MSDP direct cloud tier storage server using the latest release. Alternatively, the existing Cloud Catalyst

server can be reimaged and reinstalled as a new MSDP direct cloud tier storage server using the latest release. If you use an existing server, that server must meet the minimum requirements to be used.

See "Planning your MSDP deployment" on page 25.

Note that this operation would not be an upgrade. Instead, it would be a remove and reinstall operation. Once the new MSDP direct cloud tier storage server is available, the `nbdecommission -migrate_cloudcatalyst` utility is used to create a new MSDP direct cloud tier. This new storage can reference the data previously uploaded to cloud storage by Cloud Catalyst. When the migration process is complete and utility is run, the new MSDP direct cloud tier can be used for new backup and duplication operations. This new storage can be used for restore operations of older Cloud Catalyst images.

For more information about the `nbdecommission` command, see the NetBackup Commands Reference Guide.

The advantages of the direct migration strategy are as follows:

- Can be used for all environments including public cloud cold storage (for example: AWS Glacier or AWS Glacier Deep Archive).
- Does not require a separate server since the Cloud Catalyst server can be reimaged as an MSDP direct cloud tier server and used for migration.

The disadvantages of the direct migration strategy are as follows:

- Cannot be implemented gradually using the new MSDP direct cloud tier storage servers while Cloud Catalyst storage servers continue to be used for new backup or duplication jobs. The old Cloud Catalyst storage server cannot be used for new backup or duplication jobs while the migration process is running.
- Cloud Catalyst uses cloud storage less efficiently than MSDP direct cloud tier. This issue is especially true for NetBackup versions older than 8.2 Cloud Catalyst. This strategy continues to use existing Cloud Catalyst objects for new MSDP direct cloud tier images. Some of the cloud storage efficiency that is gained with MSDP direct cloud tier is not realized.
- Requires a new MSDP server so an existing MSDP server cannot be used and consolidation of any Cloud Catalyst servers is not possible.

See "About beginning the direct migration" on page 806.

About direct migration from Cloud Catalyst to MSDP direct cloud tiering

This section discusses the direct migration strategy to move images from a Cloud Catalyst server to MSDP direct cloud tier storage server. In this section, there are five areas that are covered:

- See "About requirements for a new MSDP direct cloud tier storage server" on page 805.
- See "About beginning the direct migration" on page 806.
- See "Placing the Cloud Catalyst server in a consistent state" on page 807.
- See "About installing and configuring the new MSDP direct cloud tier server" on page 809.
- See "Running the migration to the new MSDP direct cloud tier server" on page 810.

About requirements for a new MSDP direct cloud tier storage server

You must use a new MSDP server with no existing disk pools as the new MSDP direct cloud tier storage server for the migration. You can reinstall and reuse the Cloud Catalyst server as the new MSDP direct cloud tier server. However, it may be better to use a new MSDP server with newer hardware and keep the existing Cloud Catalyst server intact. You can keep the existing Cloud Catalyst server as a failsafe in case of an unexpected failure during the migration process.

Migration is possible to a system with less free disk space. However, an extra step is required after the creation of the new MSDP server and before you run the Cloud Catalyst migration. This extra step involves modifying the default values for `CloudDataCacheSize` and `CloudMetaCacheSize` in the `contentrouter.cfg` file.

For more information about `CloudDataCacheSize`, `CloudMetaCacheSize`, and the `contentrouter.cfg` file:

See "About the configuration items in cloud.json, contentrouter.cfg, and spa.cfg" on page 312.

You must be running the latest version of NetBackup (10.0 or later) that supports the migration feature on the new MSDP server. To do so, the primary server must also be running NetBackup 10.0 or later.

About beginning the direct migration

Determine a maintenance window during which the existing Cloud Catalyst server and the new MSDP server can be offline for the duration of the migration process. In most environments this process takes less than a day. For some very large environments or for environments with low available upload bandwidth to the cloud, the process may take longer.

Before beginning the direct migration, gather the following information:

- The Cloud Catalyst server name (hostname of Cloud Catalyst appliance or BYO server).
- The logon credentials for `root` on the Cloud Catalyst server. If the Cloud Catalyst server is an appliance, the credentials to log on and elevate the appliance to maintenance mode.
- The Cloud Catalyst storage server name (NetBackup cloud storage server that is used for Cloud Catalyst).
- The Cloud Catalyst bucket or container name.
- The KMS configuration, specifically the KMS key group name (only if KMS is configured).
  - If the Cloud Catalyst storage server type ends with `_cryptd` then KMS is enabled and `<CloudCatalyst storage server name>:<bucket/container name>` is the KMS key group name.
  - If the Cloud Catalyst storage server type ends with `_rawd` then check the `KMSOptions` section of `contentrouter.cfg` on Cloud Catalyst server. Verify if KMS is enabled and then locate the KMS key group name. If the `KMSOptions` section does not exist, then KMS is not enabled. If the `KMSOptions` section does exist, then the `KMSEnable` entry is `True` if enabled and `False` if disabled.
  - You can use the `/usr/openv/pdde/pdcr/bin/keydictutil --list` command on the Cloud Catalyst server to view these KMS settings (version 8.2 and later of Cloud Catalyst).
  - You can use the `/usr/openv/netbackup/bin/admincmd/nbkmsutil -listkgs` command on the NetBackup primary server to list the KMS key group names. Verify that the KMS key group name you have gathered exists and is correct.
- The name to be used for the new disk volume for the migrated MSDP direct cloud tier storage server.
- The name to be used for the new disk pool for the migrated MSDP direct cloud tier storage server.

- Any cloud credentials (if using AWS IAM role, plan to use the access key `dummy` and the secret access key `dummy`).
- All other cloud-specific configuration information.
- A list of all NetBackup policies and SLPs that currently write to the Cloud Catalyst storage server.

After you have gathered the previous list of information, download the `sync_to_cloud` utility from the Cohesity Download Center and make it available on the Cloud Catalyst server for use during the premigration procedure.

Verify that the MSDP data selection ID (DSID) used for Cloud Catalyst is `2`. Review the contents of the `<CloudCatalyst cache directory>/storage/databases/catalog` directory. There should be one subdirectory and the name of that subdirectory should be `2`. If there are more subdirectories or if the subdirectory `2` does not exist, contact Cohesity Support for assistance as this issue must be corrected before continuing.

On the primary server, ensure a catalog backup policy (policy type: **NBU-Catalog**) exists and it has a policy storage destination other than the Cloud Catalyst storage server to be migrated. A manual backup of this catalog backup policy is initiated at certain points in the migration process to enable rollback recovery from a failed migration. If a catalog backup on storage other than the Cloud Catalyst server does not exist, recovery from a failed migration may be difficult or impossible.

Placing the Cloud Catalyst server in a consistent state

To ensure data integrity and consistency, it is important that there are no active jobs using the Cloud Catalyst server during migration. Perform the following procedure to stop all jobs and to ensure that the Cloud Catalyst server is in a consistent and a stable state before starting the migration process.

Note: Any errors that are seen in the following procedure should be addressed before you begin the final migration. Read the full procedure and text following the procedure before you begin this process in your environment.

To place the Cloud Catalyst server in a consistent state

1. Deactivate all backup policies that write to the Cloud Catalyst storage server.
2. Deactivate all storage lifecycle policies that write to the Cloud Catalyst storage server.
3. Verify all active jobs that use the Cloud Catalyst storage server have stopped.

4 Run a catalog cleanup on the primary server using the `bpimage -cleanup` command..

Location: `/usr/openv/netbackup/bin/admincmd/bpimage -cleanup -allclients -prunetir`

5 Once the catalog cleanup completes, process the MSDP transaction queue manually on the Cloud Catalyst server using the `crcontrol --processqueue` command and wait for the processing to complete.

Location: `/usr/openv/pdde/pdcr/bin/crcontrol --processqueue`

See "Processing the MSDP transaction queue manually" on page 553.

6 Repeat step 5 to verify that all images have been processed.

7 Monitor `/usr/openv/netbackup/logs/esfs_storage` log on the Cloud Catalyst server for at least 15 minutes (at a minimum) to ensure that all delete requests have processed.

8 On the Cloud Catalyst server run the `/usr/openv/pdde/pdcr/bin/cacontrol --catalog recover all_missing` command.

Warning: If this step reports any errors, those errors must be addressed before you continue to the next step. Contact Cohesity Support if assistance is needed in addressing the errors.

---

9 On the Cloud Catalyst server run the `/usr/openv/pdde/pdcr/bin/catdbutil --list` command and redirect the output to a temporary file.

Monitor this file for errors and contact Cohesity Technical Support if any errors are reported.

10 When the previous steps have been completed without error, run the `sync_to_cloud` utility and wait for it to complete. Running this utility may take time depending on environment.

See "About beginning the direct migration" on page 806.

11 After `sync_to_cloud` has finished successfully, shut down services on the Cloud Catalyst server.

You can leave the services down on the Cloud Catalyst server. Or, if you plan to use a different MSDP server to migrate Cloud Catalyst, you can change the `Readonly` field to 1 in `<CloudCatalyst cache directory>/cache/etc/esfs.json`. Then restart the services on the Cloud Catalyst server. If the services are running on the Cloud Catalyst server at the time of migration certain configuration items like cloud bucket name are determined automatically. If not, you need to enter those configuration items you gathered in the following section:

See "About beginning the direct migration" on page 806.

12 Run a manual backup of the catalog backup policy (policy type: **NBU-Catalog**).

Do not skip this step as it is very important to run this manual backup. This backup establishes a point in time to return to if the migration does not complete successfully.

If possible, it is preferable to use a new MSDP direct cloud tier server for migration. Using a new server keeps the existing Cloud Catalyst server intact and usable if the migration unexpectedly fails. If you plan to reuse the Cloud Catalyst server as the new MSDP direct cloud tier server, you need to uninstall and or reimage the server at this time. Be sure to remove all of NetBackup and the contents of the Cloud Catalyst cache directory. If reusing a Cloud Catalyst appliance you may need to do a storage reset to remove the Cloud Catalyst cache, see the appliance documentation for details.

See "Planning your MSDP deployment" on page 25.

Note: Although not usually recommended, in some special circumstances Cloud Catalyst is running on the primary server. Since you cannot uninstall, reimage the primary server, and you cannot upgrade it with Cloud Catalyst configured, you need to run the `/usr/openv/esfs/script/esfs_cleanup.sh` script to remove Cloud Catalyst. Then you can upgrade the primary server and proceed with migration.

About installing and configuring the new MSDP direct cloud tier server

You need a new MSDP direct cloud tier server with no existing disk pools for the Cloud Catalyst migration. This section assumes that the primary server has been upgraded to the latest version of NetBackup (10.0 or later) which supports migration. Also, this section also assumes that the latest version of NetBackup (10.0 or later) has been installed on the media server or appliance to be used for migration.

Configure an MSDP direct cloud tier server on the media server to be used for migration. Do not configure any disk pools for that storage server. You must use the same KMS settings when configuring the new MSDP direct cloud tier server as were used for Cloud Catalyst. If the Cloud Catalyst storage server type ends in `_cryptd` (for example: `PureDisk_amazon_cryptd`) then KMS needs to be enabled. If the Cloud Catalyst storage server type ends in `_rawd` (for example: `PureDisk_azure_rawd`) then KMS may or may not need to be enabled. This information should be compiled before migration as noted in the *About beginning the direct migration* section.

Note: If KMS needs to be enabled then all three KMS-related checkboxes on the MSDP server configuration screen in the web UI need to be checked. Also, the KMS key group name from Cloud Catalyst needs to be entered. Mismatched KMS settings can cause problems attempting to access any of the data that Cloud Catalyst uploaded. You must verify that all KMS-related information matches.

The new MSDP direct cloud tier server must have at least 1 TB free disk space. You can migrate to a system with less free disk space. However, an extra step is required after you create the new MSDP direct cloud tier server and before you run the Cloud Catalyst migration. This extra step involves modifying the default values for `CloudDataCacheSize` and `CloudMetaCacheSize` in `contentrouter.cfg` file.

See "About the configuration items in cloud.json, contentrouter.cfg, and spa.cfg" on page 312.

The new MSDP server should be set to the correct time and you can set the time by using an NTP server. If the time is incorrect on the MSDP server, some cloud providers may report an error (for example: `Request Time Too Skewed`) and fail upload or download requests. Refer to your specific cloud vendor documentation for more information.

Note: After configuring the new MSDP server and before continuing, run a manual backup of the catalog backup policy (policy type **NBU-Catalog**). Do not skip this step as it is very important to run this manual backup. This backup establishes a point in time to return to if the migration does not complete successfully.

See "About beginning the direct migration" on page 806.

Running the migration to the new MSDP direct cloud tier server

Before you continue the process of installing and configuring the new MSDP direct cloud tier server, it is recommended that you set up logging. If any issues arise

during installation, the logs help with diagnosing any potential errors during migration. The following items are recommended:

- Ensure that the `/usr/openv/netbackup/logs/admin` directory exists before running the `nbdecommission` command.
- Set the log level to `VERBOSE=5` in the `bp.conf` file.
- Set `loglevel=3` in `/etc/pdregistry.cfg` for `OpenCloudStorageDaemon`.
- Set `Logging=full` in the `contentrouter.cfg` file.

To run the migration, go to the command prompt on the MSDP direct cloud tier server and run:

```
/usr/openv/netbackup/bin/admincmd/nbdecommission -migrate_cloudcatalyst
```

Note: This utility needs to be run in a window that does not time out or close even if it runs for several hours or more. If the migration is performed on an appliance, you need to have access to the maintenance shell and it needs to remain unlocked while the migration runs. The maintenance shell must remain enabled even if it runs for several hours or more.

Select the Cloud Catalyst storage server to migrate and enter the information as prompted by the `nbdecommission` utility.

The following is an example of what you may see during the migration:

```
# /usr/openv/netbackup/bin/admincmd/nbdecommission -migrate_cloudcatalyst
MSDP storage server to use for migrated CloudCatalyst:  myserver.test.com

Generating list of configured CloudCatalyst storage servers.
This may take a few minutes for some environments, please wait.

Cloud Storage Server Cloud Bucket CloudCatalyst Server Storage Server Type
1) amazon.com         my-bucket    myserver.test.com   PureDisk_amazon_rawd

Enter line number of CloudCatalyst server to migrate: 1

MSDP KMS encryption is enabled for amazon.com.
Please confirm that CloudCatalyst was configured using
KMSKeyGroupName amazon.com:testkey

Continue? (y/n) [n]: y

Enter new disk volume name for migrated CloudCatalyst server: newdv
```

```
Enter new disk pool name for migrated CloudCatalyst server: newdp
Enter cloud account username or access key:  AAAABBBBBCCCCCDDDDD
Enter cloud account password or
secret access key: aaaabbbbccccddddeeeeffffggg

You want to migrate amazon.com (bucket my-bucket) to
newmsdpserver.test.com (volume newdv, pool newdp).
Is that correct? (y/n) [n]: y

To fully decommission myserver.test.com after
CloudCatalyst migration is complete,run the
following command on the primary server:
/usr/openv/netbackup/bin/admincmd/nbdecommission
-oldserver myserver.test.com

Administrative Pause set for machine myserver.test.com

Migrating CloudCatalyst will include moving the images to server
newmsdpserver.test.com deleting the old disk pool, storage unit, and
storage server, deactivating policies that reference the old storage
unit, and restarting MSDP on server newmsdpserver.test.com.

Before proceeding further, please make sure that no jobs are running on
media server myserver.test.com or media server newmsdpserver.test.com.
This command may not be able to migrate CloudCatalyst
with active jobs on either of those servers.

To avoid potential data loss caused by conflicts between the
old CloudCatalyst server and the migrated MSDP server, stop the
NetBackup services on myserver.test.com if they are running.

It is recommended to make one or both of the following changes
on myserver.test.com to prevent future data loss caused by
inadvertently starting NetBackup services.
1) Rename /usr/openv/esfs/bin/vxesfsd to /usr/openv/esfs/bin/vxesfsd.off
2) Change "ReadOnly" to "1" in the esfs.json configuration file
See the documentation for more information about esfs.json.

It is also recommended to perform a catalog cleanup and backup prior
to migration so that the catalog can be restored to its original
state in the event that migration is not completed.

Continue? (y/n) [n]: y
```

```
Successfully cloned storage server: amazon.com to:
newmsdpserver.test.com_newdv

Storage server newmsdpserver.test.com has been successfully updated

The next step is to list the objects in the cloud and migrate
the MSDP catalog. The duration of this step depends on how much data
was uploaded by CloudCatalyst.
It may take several hours or longer, so please be patient.

You may reduce the duration by not migrating the
CloudCatalyst image sharing information if you are certain that
you do not use the image sharing feature.

Do you wish to skip migrating CloudCatalyst image
sharing information? (y/n) [n]:

Jun 24 15:37:11 List CloudCatalyst objects in cloud
Jun 24 15:37:13 List CloudCatalyst objects in cloud
Jun 24 15:37:18 List CloudCatalyst objects in cloud
Jun 24 15:37:26 MSDP catalog migrated successfully from CloudCatalyst

Disk pool newdp has been successfully created with 1 volumes

Moved CloudCatalyst images from myserver.test.com to newmsdpserver.test.com

Disk pool awsdp (PureDisk_amazon_rawd) is referenced by the following
storage units:
    awsdp-stu

Storage unit awsdp-stu: host myserver.test.com
Deactivating policies using storage unit awsdp-stu
Storage unit awsdp-stu is referenced by policy testaws
Deactivated policy testaws
Deleting storage unit awsdp-stu on host _STU_NO_DEV_HOST_
Deleted storage unit awsdp-stu
Deleted PureDisk_amazon_rawd disk pool awsdp
Deleted PureDisk_amazon_rawd storage server amazon.com

Stopping ocsd and spoold and spad
Checking for PureDisk ContentRouter
spoold (pid  55723) is running...
```

```
Checking for PDDE Mini SPA                                 [  OK  ]
spad (pid  55283) is running...
Checking for Open Cloud Storage Daemon                     [  OK  ]
ocsd (pid 55150) is running...
Stopping PureDisk Services
ocsd is stopped

Run MSDP utility to prepare for online checking.
This may take some time, please wait.

Starting ocsd and spoold and spad
Checking for Open Cloud Storage Daemon
ocsd is stopped
Starting Open Cloud Storage Daemon: ocsd Checking for PDDE Mini SPA
spad is stopped
spad (pid  56856) is running...                           [  OK  ]
Checking for PureDisk ContentRouter
spoold is stopped
spoold (pid  57013) is running...spoold                    [  OK  ]
Starting PureDisk Services
spoold (pid  57013) is running...

Enabling data integrity check.
Starting data integrity check.
Waiting for data integrity check to finish.
Processing the queue.
CloudCatalyst server myserver.test.com has been successfully
migrated to newmsdpserver.test.com.
To avoid potential data loss caused by conflicts between the
old CloudCatalyst server and the
migrated MSDP server, stop the NetBackup daemons (or services)
on myserver.test.com if they are running.
```

Monitor the output of the `nbdecommission` command for errors. Other logs to monitor for activity and potential errors are in the `storage_path/log/` directory. You should monitor the `ocsd_storage` log and monitor the `spad` and `spoold` logs for any `cacontrol` command issues.

If an error is encountered and you can correct the error, you can resume the migration from that point using the `start_with` option as noted in the output from the `nbdecommission` command. If you have any questions about the error, contact Cohesity Support before you resume the migration.

About the prompts during migration

During the migration, there are several prompts that are displayed when the migration is run. You can use command line options to supply answers to these prompts, if necessary. Cohesity recommends that you use the interactive prompts because it makes the migration easier to use and less error prone than using the command line options. If you choose to use the command line, the options are documented in the NetBackup Commands Reference Guide.

During the migration process most of the prompts are self-explanatory and the number and type of prompts can change. The number and type of prompts depends on the following:

- The version of Cloud Catalyst being used at time of the migration.
- If the Cloud Catalyst server is running at the time of the migration.
- If KMS is enabled on the Cloud Catalyst server.

Table B-1 discusses additional information about a few of the prompts.

Table B-1 Migration prompts

Prompts	Description
`No MSDP storage server found on myserver.test.com.` `Please create the MSDP storage server before running this utility.`	This output is displayed when the `nbdecommission -migrate_cloudcatalyst` command is run on a media server that does not have an MSDP storage server configured. See "About installing and configuring the new MSDP direct cloud tier server" on page 809.
`Disk pools exist for storage server PureDisk myserver.test.com.` `CloudCatalyst migration requires a new storage server with no configured disk pools.`	The sample output is displayed when the `nbdecommission -migrate_cloudcatalyst` command is run on a media server that does have an MSDP storage server configured and does have existing disk pools configured. Cloud Catalyst migration can only be run on a new MSDP cloud tier server with no existing disk pools.
`Enter cloud bucket name:`	If the Cloud Catalyst server is not running at the time of migration you need to manually enter the existing Cloud Catalyst bucket or container name. This information is used for migration.
`Enter CloudCatalyst server hostname:`	If the Cloud Catalyst server is not running at the time of migration you need to manually enter the server hostname of the existing Cloud Catalyst server to be migrated.

Table B-1 Migration prompts *(continued)*

Prompts	Description
`Is MSDP KMS encryption enabled for amazon.com? (y/n) [n]:`	If the Cloud Catalyst server is not running at the time of migration you may need to manually enter the KMS configuration settings for the existing Cloud Catalyst server.
`Enter new disk volume name for migrated CloudCatalyst server:`	Enter the name of the MSDP Cloud disk volume to be created on the new MSDP cloud tier server. This name is used for the migrated Cloud Catalyst data.
`Enter new disk pool name for migrated CloudCatalyst server:`	Enter the name of the MSDP Cloud disk pool to be created on the new MSDP server and used for the migrated Cloud Catalyst data.
`Enter cloud account username or access key:` `Enter cloud account password or secret access key:`	Enter the credentials for the cloud account that is used to access the Cloud Catalyst data to be migrated. If you use AWS IAM role to access the data, you should enter `dummy` for both the access key and the secret access key.

About postmigration configuration and cleanup

A successful migration results in a new disk pool for the MSDP cloud tier. If you want to use this new MSDP cloud tier server as the destination for new protection plans, policies, or duplication jobs, create a new storage unit. You must create a new storage unit for this new disk pool using the NetBackup web UI or storage API. The storage unit is not created automatically by the migration process.

Use the new storage unit as the destination for your protection plans, policies, and SLPs. You must activate any existing policies and SLPs that previously wrote to the migrated Cloud Catalyst server as the migration process disables them.

After a successful migration, you may want to clean up any obsolete objects that Cloud Catalyst created. Doing so can free up a relatively small amount of space in the cloud that is no longer needed by the MSDP cloud tier server. Cohesity recommends waiting a few days or weeks to run the `cacontrol --catalog cleanupcloudcatalystobjects` command until you are certain that the migration has been successful. After this command is run, there is no longer any possibility of reverting to Cloud Catalyst to access your data. This step is an optional and no functionality is affected if it is never done.

Run the following command to clean up the obsolete objects:

```
/usr/openv/pdde/pdcr/bin/cacontrol --catalog
cleanupcloudcatalystobjects <lsuname>
```

About the effect on image sharing

During migration, the `nbdecommission` command asks you the following question:

```
Do you wish to skip migrating CloudCatalyst
image sharing information? (y/n) [n]:
```

You can answer `y` to this question if you are certain that you do not use the image sharing feature in your NetBackup environment.

You should leave the default answer of `n` in place for all other situations or if you are unsure if your environment does not use image sharing.

You must run an additional command on the image sharing server before you can access any images that were uploaded to the cloud by Cloud Catalyst. This command should only be run if you use image sharing. Run the following command on the image sharing server:

```
/usr/openv/pdde/pdcr/bin/cacontrol
--catalog buildcloudcatalystobjects <lsuname>
```

After running the `cacontrol --catalog buildcloudcatalystobjects <lsuname>` command, restart the NetBackup services on the image sharing server.

About the effect on NetBackup Accelerator

If backups are written directly to the Cloud Catalyst server and you have the NetBackup Accelerator option enabled on your policies, there is a special consideration for Cloud Catalyst migration. The accelerator option uses the storage server name for optimization and that storage server name changes because of migration. Therefore, the first backup job that is written to the migrated MSDP cloud tier server has no accelerator optimization. Also, for accelerator enabled multiple stream policies that write directly to the migrated MSDP cloud tier server, the deduplication rate may be zero for the first backup job. Subsequent backup jobs return to normal accelerator optimization and deduplication rates.

The migration has no effect on NetBackup Accelerator enabled policies if those policies write to MSDP and then use a duplication job to write to Cloud Catalyst.

About the effect on the `MachineState` setting

The `nbdecommission` command sets `MachineState` to `administrative pause (13)` for some servers. When a server has the setting `MachineState` set to `administrative pause (13)`, no jobs run, and the server may appear down.

You can display `MachineState` with the following command:

```
/usr/openv/netbackup/bin/admincmd/nbemmcmd -listhosts
-display_server -machinename myserver.test.com
-machinetype media -verbose
```

If you need to clear the `administrative pause MachineState` for a server, run the following command:

```
/usr/openv/netbackup/bin/admincmd/nbemmcmd -updatehost
-machinename myserver.test.com -machinetype media
-machinestateop clr_admin_pause -masterserver mymaster.test.com
```

About the Cloud Catalyst migration `-dryrun` option

The `-dryrun` option can be added to the `nbdecommission -migrate_cloudcatalyst` command. The `-dryrun` may be useful in some environments as a test run for the migration. The `-dryrun` option does not perform all migration steps and so a successful execution with this option does not guarantee success when the actual migration is attempted. This option is useful to identify any errors that can be addressed before the actual migration.

The `-dryrun` option creates the new MSDP cloud tier server and migrates the Cloud Catalyst data. Then it deletes the newly added MSDP cloud tier server before your environment is returned to the previous state.

Note: The `-dryrun` option does not modify the primary server catalog entries to move the images to the new MSDP cloud tier server. Therefore, you cannot do a test restore or other operations to access the data when using the `-dryrun` option.

After using the `-dryrun` option you must manually delete the newly added cloud volume in the cloud storage (for example: AWS, Azure, or other cloud vendor) using the cloud console or other interface. If you do not delete this new volume, then future migration operations are affected.

About Cloud Catalyst migration `cacontrol` options

NetBackup has multiple `cacontrol` options that help with cleanup of images and help to make the Cloud Catalyst migration more successful.

Note: Multiple `cacontrol` command options are not intended to be run directly because running the `nbdecommision` command activates the `cacontrol` option. Carefully review all options in Table B-2.

Table B-2 lists the `cacontrol` command options that you can use during the Cloud Catalyst migration and how to use those options.

Table B-2 `cacontrol` options

`cacontrol` option	**Description**
`buildcloudcatalystobjects`	Location: `/usr/openv/pdde/pdcr/bin/cacontrol --catalog buildcloudcatalystobjects <lsuname>` `<lsuname> = Name of the MSDP Cloud LSU that was migrated from CloudCatalyst.` This option creates a lookup table for image sharing after successful migration to the MSDP cloud tier. After migration, this command should be run on the image sharing server and then the services on that server should be restarted.
`cleanupcloudcatalystobjects`	Location: `/usr/openv/pdde/pdcr/bin/cacontrol --catalog cleanupcloudcatalystobjects <lsuname>` `<lsuname> = Name of the MSDP Cloud LSU that was migrated from CloudCatalyst.` This option removes unused Cloud Catalyst objects from the cloud after successful migration to the MSDP cloud tier server. This command can be run as an optional step which may be run a few days or weeks after the migration. This option cleans up any Cloud Catalyst objects which the new MSDP cloud tier server does not need. Do not run unless confident that the migration was successful since you cannot revert to Cloud Catalyst to access the data once this command is run.

Table B-2 `cacontrol` options *(continued)*

`cacontrol` option	Description
`migratecloudcatalyst`	Location: `/usr/openv/pdde/pdcr/bin/cacontrol --catalog migratecloudcatalyst <lsuname> <cloudcatalystmaster> <cloudcatalystmedia> [skipimagesharing] [start_with]` `<lsuname> = Name of the MSDP Cloud LSU to be migrated from CloudCatalyst.` `<cloudcatalystmaster> = Master server name.` `<cloudcatalystmedia> = Media server hostname of the CloudCatalyst server to be migrated.` `[skipimagesharing] = Flag which indicates to skip migrating the image sharing data from CloudCatalyst to the new MSDP Cloud LSU.` `[start_with] = Indicates the point at which to resume a failed migration after the cause of the failure has been addressed.` The `nbdecommission -migrate_cloudcatalyst` command calls this `cacontrol` command as needed. Do not run this `cacontrol` directly. Instead, use the `nbdecommission -migrate_cloudcatalyst` command to perform the migration.
`migratecloudcatalyststatus`	Location: `/usr/openv/pdde/pdcr/bin/cacontrol --catalog migratecloudcatalyststatus <lsuname>` `<lsuname> = Name of the MSDP Cloud LSU being migrated from CloudCatalyst.` The `nbdecommission -migrate_cloudcatalyst` command calls this `cacontrol` command as needed. Do not run this `cacontrol` directly. Instead, use the `nbdecommission -migrate_cloudcatalyst` command to perform migration.

Reverting back to Cloud Catalyst from a successful migration

The process to revert back to Cloud Catalyst assumes that a NetBackup catalog backup was performed for the primary server catalog before the `nbdecommission -migrate_cloudcatalyst` command was run. If no such NetBackup catalog backup image is available, it is not possible to revert to Cloud Catalyst because the migration process modifies the NetBackup catalog.

The reversion process also assumes that the command `/usr/openv/pdde/pdcr/bin/cacontrol --catalog cleanupcloudcatalystobjects` has not been run on the migrated MSDP cloud tier server. The reason for that is because once that command has been run, it is not possible to revert back to Cloud Catalyst.

The images that Cloud Catalyst wrote and that have expired since the migration was completed, have been removed from the cloud storage. Reverting to Cloud Catalyst does not make these images available for restore as that data no longer exists.

All caveats and limitations of performing a NetBackup primary server catalog recovery apply, see the section of the NetBackup admin guide that discusses catalog recovery in detail. Specifically, no data is written to the MSDP server or other storage servers after the point in time at which the catalog backup image was created is available. The data is not available for a restore after the NetBackup primary server catalog recovery is performed.

You can use one of the following procedures to revert back to Cloud Catalyst:

- Reverting back to Cloud Catalyst when the server is in the same state when the migration was performed
- Reverting back to Cloud Catalyst when the server was reused and or reinstalled when the migration was performed

The following procedure assumes that the Cloud Catalyst server has been left in the same state that it was in at the time of migration and all services are stopped.

Reverting back to Cloud Catalyst when the server is in the same state when the migration was performed

1. Stop the NetBackup services on the new MSDP cloud tier server.
2. Open the **NetBackup web UI**.
3. Click **Recovery**. Then click **NetBackup catalog recovery**.

4. Select the catalog backup image that was created before running the `nbdecommission -migrate_cloudcatalyst` command to migrate Cloud Catalyst to MSDP cloud tier server.
5. Complete all steps in the UI to recover the NetBackup catalog.
6. Stop and restart the NetBackup services on the primary server.
7. On the Cloud Catalyst server, ensure that the `esfs.json` file has the setting `ReadOnly` set to `0`.

   If you only need to do restores and do not intend to run new backup or duplication jobs to Cloud Catalyst, then set `ReadOnly` to `1`.
8. Start the NetBackup services on the Cloud Catalyst server.
9. Once the Cloud Catalyst storage server has come online, you can proceed with restores, backups, or optimized duplication jobs.

   Backup or optimized duplication jobs require that `ReadOnly` is set to `0` in the `esfs.json` file.
10. If running a Cloud Catalyst version older than 8.2 (example: 8.1, 8.1.1, 8.1.2), you may need to deploy a new host name-based certificate for the media server. You can deploy the certificate by running the following command on the primary server:

    ```
    /usr/openv/netbackup/bin/admincmd/bpnbaz -ProvisionCert
    <CloudCatalyst host-name>
    ```

    You must restart the NetBackup services on the Cloud Catalyst server.
11. You may need to run the following command to allow Cloud Catalyst to read from the bucket in the cloud storage:

    ```
    /usr/openv/esfs/bin/setlsu_ioctl
    <cachedir>/storage/proc/cloud.lsu <bucketname>
    ```

    No harm is done if you run this command when it is not needed. If you do run the command, you can see the following output:

    ```
    return code: -1

    File exists.
    ```

12. (Optional) Remove the entire MSDP cloud sub-bucket folder in cloud storage to avoid wasted space and avoid any problems with future migration to MSDP cloud tier server.

The following procedure assumes that the Cloud Catalyst server was reused and or reinstalled as an MSDP cloud tier server or is unavailable for some other reason.

Reverting back to Cloud Catalyst when the server was reused and or reinstalled when the migration was performed

1. Stop the NetBackup services on the new MSDP cloud tier server.
2. Open the **NetBackup web UI**.
3. Click **Recovery**. Then click **NetBackup catalog recovery**.
4. Select the catalog backup image that was created before running the `nbdecommission -migrate_cloudcatalyst` command to migrate Cloud Catalyst to MSDP cloud tier server.
5. Complete all steps in the UI to recover the NetBackup catalog.
6. Stop and restart the NetBackup services on the primary server.
7. Reinstall the Cloud Catalyst server using the same NetBackup version and EEB bundles that were active when migration was performed.
8. Then contact Cohesity Technical Support to use the `rebuild_esfs` process to recover that Cloud Catalyst server from the data in cloud storage. (The `rebuild_esfs` process supersedes the old `drcontrol` method of recovering a Cloud Catalyst server. The `drcontrol` method is deprecated.)
9. (Optional) Remove the entire MSDP cloud sub-bucket folder in cloud storage to avoid wasted space and avoid any problems with future migration to MSDP cloud tier server.

Reverting back to Cloud Catalyst from a failed migration

Recovering the NetBackup primary server catalog to revert back to Cloud Catalyst is the safest approach for both successful and a failed migration attempts. However, it may be possible to revert to Cloud Catalyst from a failed migration attempt without recovering the primary server catalog.

If the failure occurred and the `nbdecommission` command exits before displaying the following message, then you may be able to revert back to Cloud Catalyst without recovering the primary server catalog. The following message is displayed in the output from the command or the `admin` log file for the `nbdecommission` command:

```
Disk pool <new disk pool name> has been successfully
created with 1 volumes
```

Migration failures that occur after the `Disk pool` message is displayed require recovering the primary server catalog to revert to Cloud Catalyst.

If you do not recover the primary server catalog, you must manually delete the new disk pool, disk volume, cloud storage server, and the MSDP cloud tier server. You must delete these after reverting back to Cloud Catalyst.

The following procedure assumes that the migration fails before the `Disk pool` message appears in the output. The procedure also assumes that the Cloud Catalyst server is not reused as the MSDP cloud tier server for migration.

Reverting back to Cloud Catalyst after a failed migration

1. Stop the NetBackup services on the new MSDP cloud tier server.
2. On the Cloud Catalyst server, ensure that the `esfs.json` file has `ReadOnly` set to `0`.

   If you only need to do restores and do not intend to run new backup or duplication jobs to Cloud Catalyst, then set `ReadOnly` to `1`.
3. Start the NetBackup services on the Cloud Catalyst server.
4. Once the Cloud Catalyst storage server has come online, you can proceed with restores, backups, or optimized duplication jobs.

   Backup or optimized duplication jobs require that `ReadOnly` is set to `0` in the `esfs.json` file.
5. If running a Cloud Catalyst version 8.2 or earlier, you may need to deploy a new host name-based certificate for the media server. You can deploy the certificate by running the following command on the primary server:

   ```
   /usr/openv/netbackup/bin/admincmd/bpnbaz -ProvisionCert
   <CloudCatalyst host-name>
   ```

   You must restart the NetBackup services on the Cloud Catalyst server.

6 You may need to run the following command to allow Cloud Catalyst to read from the bucket in the cloud storage:

```
/usr/openv/esfs/bin/setlsu_ioctl
<cachedir>/storage/proc/cloud.lsu <bucketname>
```

No harm is done if you run this command when it is not needed. If you do run the command, you can see the following output:

```
return code: -1

File exists.
```

7 (Optional) Remove the entire MSDP cloud sub-bucket folder in cloud storage to avoid wasted space and avoid any problems with future migration to MSDP cloud tier server.

The following procedure assumes that the migration fails on the Cloud Catalyst server that was reused and or reinstalled as an MSDP cloud tier server.

Reverting back to Cloud Catalyst after a failed migration when the Cloud Catalyst server was reused

1 Stop the NetBackup services on the new MSDP cloud tier server.

2 Reinstall the Cloud Catalyst server using the same NetBackup version and EEB bundles that were active when migration was performed.

3 Then contact Cohesity Technical Support to use the `rebuild_esfs` process to recover that Cloud Catalyst server from the data in cloud storage. (The `rebuild_esfs` process supersedes the old `drcontrol` method of recovering a Cloud Catalyst server. The `drcontrol` method is deprecated.)

4 (Optional) Remove the entire MSDP cloud sub-bucket folder in cloud storage to avoid wasted space and avoid any problems with future migration to MSDP cloud tier server.

Appendix C

Encryption Crawler

This appendix includes the following topics:

- About the Encryption Crawler
- About the two modes of the Encryption Crawler
- Managing the Encryption Crawler
- Advanced options
- Tuning options
- Encrypting the data
- Command usage example outputs
- Updating the KMS configuration

About the Encryption Crawler

The Encryption Crawler searches all MSDP pools to check from unencrypted data. It traverses all the existing data containers and if a data segment is not encrypted, that segment is encrypted with AES-256-CTR algorithm. The Encryption Crawler encrypts the encryption keys of any data segments the KMS automatic conversion process has not processed if KMS is enabled. The KMS automatic conversion process encrypts the encryption keys of all the existing encrypted data.

See “About MSDP Encryption using NetBackup Key Management Server service” on page 114.

Several conditions may lead to an MSDP pool having unencrypted data segments even though the user intends to encrypt all data:

- Encryption is not enabled when the pool is configured. Encryption is only enabled after backup data is ingested into the pool.

- The `encrypt` keyword is not added to the `ServerOptions` option in `contentrouter.cfg` of the MSDP. In this case, encryption is not enabled for all `pd.conf` that may exist on the MSDP host, load-balancing media servers, build-your-own (BYO) servers, and NetBackup Client Direct.

Late backups may reference the unencrypted data and may not go away when the old images expire. The Encryption Crawler is used to encrypt all the existing data residing in an MSDP pool which was not previously encrypted.

The Encryption Crawler requires that encryption is properly configured. The `encrypt` keyword is required to be added to the `ServerOptions` option in `contentrouter.cfg` for the MSDP pool. If an Instant Access or Universal Share is configured, Encryption Crawler requires that encryption is enabled for VpFS. Additionally, you must create all the checkpoints for all the existing VpFS shares after encryption is enabled. If the environments are upgraded from a release before NetBackup 8.1, the Encryption Crawler requires all rolling data conversion processes finish.

About the two modes of the Encryption Crawler

The Encryption Crawler is not turned on by default. You must explicitly enable it with the `crcontrol` command. Encryption Crawler has two modes: **Graceful** mode and **Aggressive** mode. These two modes can have an effect on how certain jobs perform. Review the following information to help you select the right mode for your environment.

Graceful mode

Unless the user specifies a different mode with the `crcontrol --encconvertlevel` command, Encryption Crawler's default mode is **Graceful**. In this mode, it runs only when the MSDP pool is relatively idle and no compaction or CRQP jobs are active. It usually means no backup, restore, duplication, or replication jobs are active on the MSDP pool when the MSDP pool is idle. To prevent Encryption Crawler from overloading the system it doesn't run continuously. When the Encryption Crawler is in **Graceful** mode, it may take a longer time to finish.

The **Graceful** mode checks that the MSDP pool is relatively idle. It checks the pool state by calculating the I/O statistics on the MSDP pool and checks that no compaction or CRQP jobs are active before it processes each data container. It pauses if the MSDP pool is not idle, compaction, or CRQP jobs are active. In most cases, **Graceful** mode pauses when backup, restore, duplication, or replication jobs are active on the MSDP pool.

If the data deduplication rate of the active NetBackup jobs is high, the I/O operation rate could be low and the MSDP pool could be relatively idle. In this case, the **Graceful** mode may run if no compaction or CRQP jobs are active.

If the MSDP fingerprint cache loading is in progress, the I/O operation rate on the MSDP pool is not low. In this case, the **Graceful** mode may pause and wait for the fingerprint cache loading to finish. The Encryption Crawler monitors the `spoold` log and waits for the message that begins with `ThreadMain: Data Store nodes have completed cache loading` before restarting. The location of the `spoold` log is: *`storage_path`*`/log/spoold/spoold.log`. To check if compaction or CRQP jobs are active, run the `crcontrol --compactstate` or `crcontrol --processqueueinfo` command.

To have the **Graceful** mode run faster, you can use the Advanced Options of `CheckSysLoad`, `BatchSize`, and `SleepSeconds` to tune the behavior and performance of **Graceful** mode. With a larger number for `BatchSize` and a smaller number for `SleepSeconds`, **Graceful** mode runs more continuously.

If you turn off `CheckSysLoad`, **Graceful** mode runs while backup, restore, duplication, replication, compaction, or CRQP jobs are active. Such changes can make **Graceful** mode more active, however it's not as active as **Aggressive** mode.

Aggressive mode

In this mode, the Encryption Crawler disables CRC check and compaction. It runs while backup, restore, duplication, replication, or CRQP jobs are active.

The **Aggressive** mode affects the performance of backup, restore, duplication, and replication jobs. To minimize the effect, use the **Graceful** mode. This choice temporarily pauses the encryption process when the system is busy and can slow down that process. The **Aggressive** mode keeps the process active and aggressively running regardless of system state.

The following points are items to consider when **Aggressive** mode is active:

- Any user inputs and the last progress are retained on MSDP restart. You don't need to re-run the command again to recover. The Encryption Crawler recovers and continues from the last progress automatically.
- You must enforce encryption with the `encrypt` keyword on the `ServerOptions` option in the `contentrouter.cfg` file in MSDP. You must also restart MSDP before enabling Encryption Crawler, otherwise the Encryption Crawler does not indicate that it is enabled.
- If your environment is upgraded from a release older than NetBackup 8.1, you must wait until the rolling Data Conversion finishes before you enable the Encryption Crawler. If you don't wait, the Encryption Crawler does not indicate that it is enabled.
- You cannot repeat the Encryption Crawler process after it finishes. Only the data that existed before you enable encryption is unencrypted. All the new data is encrypted inline and does not need the scanning and crawling.

- If you disable encryption enforcement after the Encryption Crawler process finishes, the Encryption Crawler state is reset. You can restart the Encryption Crawler process when encryption is enforced again. The time that is required to finish depends on the following items:
  - How much new and unencrypted data is ingested.
  - How much data resides on the MSDP pool.

Resource utilization for the Graceful and Aggressive modes

Memory: The Encryption Crawler can consume an additional 1 GB of memory for each MSDP partition. The **Graceful** mode consumes less memory than the **Aggressive** mode.

CPU: The major CPU utilization by the Encryption Crawler is by the data encryption with AES-256-CTR algorithm. The CPU utilization is less than backing up the same quantity of data. During the process, there is no fingerprinting, inter-component, or inter-node data transfer happening.

Disk I/O: The Encryption Crawler is I/O intensive especially in the **Aggressive** mode. The **Aggressive** mode competes for I/O significantly with the active jobs, and it may commit more I/O than the backup jobs.

Managing the Encryption Crawler

Use the `crcontrol` command to manage the Encryption Crawler. The following table describes the options you can use to manage how the Encryption Crawler functions.

Table C-1 `crcontrol` command options

Option	Description
`--encconverton`	To enable and start the Encryption Crawler process, use `--encconverton [num]`. The *num* variable is optional and indicates the number for the partition index (starting from 1). The parameter enables the Encryption Crawler for the specified MSDP partition. If *num* is not specified, it is enabled for all MSDP partitions. The *num* variable is not supported on a BYO setup when the `/etc/nbapp-release` (Linux), or `c:\etc\nbapp-release` (Windows) file isn't present. On a BYO setup, create the file to enable multiple volumes support and then the *num* variable is supported. For example, `[root@vrawebsrv4663 ~]# /usr/openv/pdde/pdcr/bin/crcontrol --encconverton Encryption conversion turned on for all partitions` See "About provisioning the storage for MSDP" on page 57.
`--encconvertoff`	To disable and stop the Encryption Crawler process, use `--encconvertoff [num]`. The *num* variable is optional and indicates the number for the partition index (starting from 1). The parameter enables the Encryption Crawler for the specified MSDP partition. If *num* is not specified, it disabled for all MSDP partitions. The *num* variable is not supported on a BYO setup when the `/etc/nbapp-release` (Linux), or `c:\etc\nbapp-release` (Windows) file isn't present. On a BYO setup, create the file to enable multiple volumes support and then the *num* variable is supported. See "About provisioning the storage for MSDP" on page 57.

Table C-1 `crcontrol` command options *(continued)*

Option	Description
`--encconvertlevel`	To switch between **Graceful** mode and **Aggressive** mode, use `--encconvertlevel level`. The *`level`* is required. ■ A value of 1 for the *`level`* variable is the default for **Graceful** mode. For example, `/usr/openv/pdde/pdcr/bin/crcontrol --encconvertlevel 1` ■ A value for the *`level`* variable that is between 2-4 indicates that **Aggressive** mode is enabled. A larger number indicates that the Encryption crawler is more aggressive. For example, `/usr/openv/pdde/pdcr/bin/crcontrol --encconvertlevel 2`
`--encconvertstate`	To determine the mode of the Encryption Crawler process and the progress, use `--encconvertstate [verbose]`. Optionally, you can specify a verbose level (0-2) for this option. ■ 0 is the default verbose level for the overall brief information. ■ 1 is for the overall information and the details of each partition. ■ 2 is for the overall information and the details of each partition. The details of a partition are shown even if the process is finished for the partition. The `verbose` parameter is not supported on a BYO setup when the `/etc/nbapp-release` (Linux), or `c:\etc\nbapp-release` (Windows) file isn't present. On a BYO setup, create the file to enable multiple volumes support and then the *`num`* variable is supported. For example, `/usr/openv/pdde/pdcr/bin/crcontrol --encconvertstate 2` See "About provisioning the storage for MSDP" on page 57.

Once the Encryption Crawler is turned on, you can monitor the status, mode, and progress with the `crcontrol --encconvertstate` command.

Table C-2 Encryption Crawler monitor

Item	Description
Status	Shows if the Encryption Crawler is **ON**, **OFF**, or **Finished**.
Level	Shows in which level and mode the Encryption Crawler is. The value is in the format *mode* (*level*), for example **Graceful (1)**.
Busy	Shows if the Encryption Crawler is busy or not.
Max Group ID	The maximum container group ID to process when the Encryption Crawler is turned on. It's the data boundary and doesn't change once Encryption Crawler is turned on.
Current Group ID	Currently processing this group ID.
Current Container ID	Currently processing this container ID.
Containers Estimated	The estimated number of data containers in the MSDP pool that the Encryption Crawler must process. It's a statistic information and there may be inaccuracy for performance reasons. Once the Encryption Crawler is turned on, the value is not updated.
Containers Scanned	The number of data containers the Encryption Crawler must process.
Containers Converted	The number of containers encrypted by the Encryption Crawler process.
Containers Skipped	The number of data containers that the Encryption Crawler skipped. The reasons vary and are described in About the skipped data containers. If there are skipped data containers, you can check the Encryption Crawler log or the history log for the details. The `encryption_reporting` tool may help report and encrypt the individual containers after the Encryption Crawler process finishes. Details about this `encryption_reporting` tool are available. See "Encrypting the data" on page 839. See "Command usage example outputs" on page 840.
Data Size Scanned	The aggregated data size of the scanned data containers for **Containers Scanned**.

Table C-2 Encryption Crawler monitor *(continued)*

Item	Description
Data Size Converted	The aggregated data size of the converted data containers for **Containers Converted**.
Progress	The proportion of the total estimated data containers that the Encryption Crawler has scanned. **Progress = Containers Scanned / Containers Estimated**
Conversion Ratio	The proportion of the scanned data size which the Encryption Crawler has converted. **Conversion Ratio = Data Size Converted / Data Size Scanned**
Mount Points Information	The status of each mount point. If a verbose value of 1 is specified for the `--encconvertstate` option, the details of the unfinished mount points are printed. If a verbose value of 2 is specified for `--encconvertstate` option, the details of all the mount points are printed regardless of completion state.

The **Progress** line in the log can be used to extrapolate how long the Encryption Crawler is expected to take. For example, if 3.3% of the pool is completed in 24 hours, the process may take about 30 days to finish.

Note: The Encryption Crawler processes the data containers in reverse order from new to old.

It's possible to back up new data after encryption is enforced but before the Encryption Crawler is turned on. If that happens, the **Conversion Ratio** could be less than 99% for the new data containers at the beginning. While the process is running, the value of **Conversion Ratio** can become higher with the fact that the older data containers can potentially have more unencrypted data. In this case, the **Conversion Ratio**, **Containers Converted**, and **Containers Estimated** can help estimate the speed for these data containers.

Monitoring the change of **Conversion Ratio** can give some indication for the proportion of the unencrypted data while the Encryption Crawler is active.

Note: During the encryption process, the progress survives in the case of MSDP restart.

About the skipped data containers

The reasons the Encryption Crawler skips some data containers as reported by **Containers Skipped** include:

- If a data container is to be expired but not yet deleted, it is skipped.
- If a data container has a possible data integrity issue, it is skipped. The Encryption Crawler conveys the container to the CRC check process to identify and possibly fix the container.
- If Instant Access or Universal Share is configured, and if some shares are not checkpointed before the Encryption Crawler process, the shares may hold some data containers with exclusive permission. Those data containers are skipped. Cohesity recommends that you create checkpoints for all the shares of Instant Access or Universal Share before turning on the Encryption Crawler process. By doing so, VpFS releases the exclusive permission of those data containers for `spoold` and the Encryption Crawler to process them.
- Appliances starting with the release of 3.1.2 can have empty data containers the VpFS root share `vpfs0` reserves, even if Instant Access or Universal Share is configured. This situation can also occur on a BYO setup where Instant Access or Universal Share is configured. Normally, VpFS does not release the exclusive permission of those data containers. Those data containers are skipped. You can ignore these skipped containers.

  Here how to check if the skipped data containers are empty and if the VpFS root share `vpfs0` owns them. You can check the other VpFS owned data containers in the similar way.

  - You can find the skipped data containers that are identified as owned by VpFS in the Encryption Crawler log by looking for the following:

    ```
    n152-h21:/home/maintenance # grep VpFS
    /msdp/data/dp1/pdvol/log/spoold/enccrawler.log
    February 04 05:13:14 WARNING [139931343951616]: -1:
    __getDcidListFromOneGroup: 1 containers owned by VpFS in group
    7 were skipped. min DC ID 7168, max DC ID 7168
    ```

  - Check if the VpFS root share `vpfs0` owns the data containers.

```
n152-h21:/home/maintenance # cat /msdp/data/dp1/4pdvol/7/.shareid
vpfs0
106627568
```

- The data containers that the VpFS root share `vpfs0` owns, are empty.

```
n152-h21:/home/maintenance # ls -Al /msdp/data/dp1/4pdvol/7
total 24
-rw-r--r-- 1 root root 64 Feb 1 02:40 7168.bhd
-rw-r--r-- 1 root root 0 Feb 1 02:40 7168.bin
-rw------- 1 root root 12 Feb 1 02:40 .dcidboundary
-rw-r----- 1 root root 15 Feb 1 02:40 .shareid
drwxr-xr-x 3 root root 96 Feb 4 15:37 var
n152-h21:/home/maintenance # /usr/openv/pdde/pdcr/bin/dcscan 7168
Path = /msdp/data/dp1/4pdvol/7/7168.[bhd, bin]
*** Header for container 7168 ***
version : 1
flags : 0x4000(DC_ENTRY_SHA256)
data file last position : 0
header file last position : 64
source id : 0
retention : 0
file size : 0
delete space : 0
active records : 0
total records : 0
deleted records : 0
crc32 : 0x1d74009d
```

Advanced options

You can specify the options that are shown under the **EncCrawler** section in `contentrouter.cfg` to change the default behavior of the Encryption Crawler. The options only affect the **Graceful** mode and these options don't exist by default. You must add them if needed.

After you change any of these values, you must restart the Encryption Crawler process for the changes to take effect. Restart the Encryption Crawler process with the `crcontrol` command and the `--encconvertoff` and `--encconverton` options. You do not need to restart the MSDP services.

After the initial tuning, you may want to occasionally check the progress and the system effect for the active jobs. You can do further tuning at any point during the process if desired.

Table C-3 Advanced options

Option	Value	Description
`SleepSeconds`	**Type:** Integer **Range:** 1-86400 **Default:** 5	This option is the idle time for the **Graceful** mode after it processes a batch of data containers. The default setting is 5 seconds and the range is 1-86400 seconds.
`BatchSize`	**Type:** Integer **Range:** 1-INT_MAX **Default:** 20	This option is the data container number for which the **Graceful** mode processes as a batch between the idle time. The default setting is 20.
`CheckSysLoad`	**Type:** Boolean **Range:** yes or no **Default:** yes	The **Graceful** mode does not run if it detects an active backup, restore, duplication, replication, compaction, or CRQP job. When you set this option to `no`, the **Graceful** mode does not do the checking. Instead, it processes a number of `BatchSize` data containers, then sleeps for a number of `SleepSeconds` seconds, then processes another batch and then sleeps. It continues this process until complete.

Tuning options

Tuning the Graceful mode

To have a faster **Graceful** mode, one can leverage the `CheckSysLoad`, `BatchSize`, and `SleepSeconds` options to tune the behavior and performance of the **Graceful** mode.

See "Advanced options" on page 835.

With a larger number for `BatchSize` and a smaller number for `SleepSeconds`, the **Graceful** mode runs more continuously. When you turn off `CheckSysLoad`, the **Graceful** mode keeps running while backup, restore, duplication, replication, compaction, or CRQP jobs are active. Such changes can make the **Graceful** mode more aggressive, although not as aggressive as the **Aggressive** mode. The advantage is the tuned **Graceful** mode has less effect on the system performance than the **Aggressive** mode for backup, restore, duplication, and replication jobs. It has even less effect than the **Aggressive** mode with the lowest level 2. The trade-off, especially when `CheckSysLoad` is turned off, is that it becomes semi-aggressive. It can affect the system performance for the active jobs and it makes the CRC check, CRQP processing, or compaction take a longer time to run and finish.

Tuning the Aggressive mode

Aggressive mode has three levels, 2-4. The higher level means more aggressive and usually better performance for the Encryption Crawler. It also means more effect on the system performance for backup, restore, duplication, replication jobs.

For the best performance of the Encryption Crawler, use Level 2-4 for the **Aggressive** mode based on the daily system loads. Otherwise, use Level 1 for the **Graceful** mode. Please note that the **Aggressive** mode with a higher level doesn't result in a better overall system performance for both the Encryption Crawler and the active jobs. It doesn't mean that the **Aggressive** mode performs better than the **Graceful** mode either. You may need to monitor the progress of the Encryption Crawler and the system effect for the active jobs to find the best fit.

You can consider dynamically switching between the **Aggressive** mode and the **Graceful** mode for a period of a half day to multiple days. Make the changes according to the pattern of the daily system loads and active jobs. Dynamically switching helps you to discover which mode works for your environment.

See "Managing the Encryption Crawler" on page 829.

See "About the two modes of the Encryption Crawler" on page 827.

Turn on Encryption Crawler for part of the MSDP partitions to reduce system effect

The **Aggressive** mode affects the performance of backup, restore, duplication, and replication jobs. The tuned **Graceful** mode does as well, although not as seriously as the **Aggressive** mode. To reduce the system effect, one can selectively have the Encryption Crawler turned on for part of the MSDP partitions at the same time.

Selectively disable DataStore Write for the MSDP partitions to reduce system effect

The **Aggressive** mode affects the performance of backup, restore, duplication, and replication jobs. The tuned **Graceful** mode does as well, although not as seriously as the **Aggressive** mode. To reduce the system effect, you can selectively disable DataStore Write for the MSDP partitions which have Encryption Crawler running. It can be done with the `crcontrol --dswriteoff` command for BYO setup. For a NetBackup appliance, the command should be executed through the CLISH. Otherwise the NetBackup appliance resets the state automatically after a short time.

You must reset the DataStore Write state when the process finishes to allow the partitions to take in new backup data.

Tuning recommendations for the Encryption Crawler

Table C-4 Tuning recommendations

Actions	Explanation
Turn on Encryption Crawler in the **Graceful** mode with the default settings.	Cohesity recommends that you wait for fingerprint cache loading to complete before you perform any backups or turn on the Encryption Crawler. Determine when to start by monitoring the `spoold` log and waiting for the message that begins with `ThreadMain: Data Store nodes have completed cache loading`. The Encryption Crawler is in the **Graceful** mode by default when you start it. After you start Encryption Crawler, allow it to run for 24 hours to 48 hours with normal backup, duplication, and replication jobs. After this time, the progress of Encryption Crawler process can be checked with the `crcontrol --encconvertstate` command. After you check on the Encryption Crawler process, review the following: First, check the **Progress** item and confirm Encryption Crawler progress. If there is no progress or not in the expected speed, you need to make changes to make faster process. Use the **Progress** item to extrapolate how long Encryption Crawler is expected to take. For example, if 3.3% of the pool is completed in 24 hours, the process may take about 30 days to finish. If the speed is slower than desired, make adjustments to make the Encryption Crawler faster as shown in this process. Please note the Encryption Crawler processes the data containers in reverse order from new to old. It's possible to back up new data after encryption is enforced but before the Encryption Crawler is turned on. If that happens, the **Conversion Ratio** could be less than 99% for the new data containers at the beginning. While the process is active, the value of **Conversion Ratio** can become higher with the fact that the older data containers can potentially have more unencrypted data. In this case, the **Conversion Ratio**, **Containers Converted**, and **Containers Estimated** can give more hints to determine the speed for these data containers. Monitoring the change of **Conversion Ratio** can give some hints on the proportion of the unencrypted data while the Encryption Crawler is active. See "Managing the Encryption Crawler" on page 829.

Table C-4 Tuning recommendations *(continued)*

Actions	Explanation
Tune the **Graceful** mode to run faster.	You can use the information in Tuning the Graceful mode to speed up the **Graceful** mode. After the initial tuning, you may need to check the progress and the system effect for the active jobs occasionally. You can do further tuning at any point during the process if desired. If the tuned **Graceful** mode negatively affects the system performance for the active jobs, you can consider turning off the Encryption Crawler for some of the MSDP partitions. You can keep it running for other partitions by following the recommendations in Turn on Encryption Crawler for part of the MSDP partitions to reduce system effect to reduce the system effect. You can also consider turning off the DataStore Write permission for some MSDP partitions by following the recommendations in Selectively disable DataStore Write for the MSDP partitions to reduce system effect which have the Encryption Crawler running. If the processing speed doesn't meet the expectations, the **Aggressive** mode can be leveraged for your environment.
Turn on the **Aggressive** mode.	You can use the information in Tuning the Aggressive mode to have the best performance for the Encryption Crawler. Cohesity recommends that you start from the lowest level 2, then gradually increase to a higher level. You may need to check the progress and the system effect for the active jobs occasionally. You can perform further tuning at any point during the process if desired.
Find the tuning point which best balances the process speed and the system effect.	A faster Encryption Crawler speed usually means more effect on the system for all active jobs. A combination of tuning options may contribute a good balance between both.

Encrypting the data

This procedure shows you how to encrypt all your MSDP data. Be aware you can run the `encryption_reporting` tool in step 4 at any time. It's an independent tool that is used to report the unencrypted data.

Encrypting all MSDP data

1. Enforce encryption in MSDP if it's not enforced.

   Add the `encrypt` keyword to the `ServerOptions` option in `contentrouter.cfg`, and restart MSDP to enforce encryption. Please ensure that no conflict or duplicate keywords are present before adding it. A conflict keyword is `noencrypt`. For the details of enabling or enforcing encryption, please refer to the following:

   See "About MSDP encryption" on page 109.

   If Instant Access or Universal Share is configured, you must change `vpfsd_config.json` and restart VpFS to enable encryption separately. You must also create checkpoints for all the VpFS shares after encryption is enabled.

2. If the rolling data conversion is in progress, wait until it finishes.

3. Run the Encryption Crawler process until it finishes.

   More information about how to run, tune, and monitor the progress of Encryption Crawler is available.

   See "About the two modes of the Encryption Crawler" on page 827.

   See "Managing the Encryption Crawler" on page 829.

   See "Tuning options" on page 836.

4. Run the reporting tool `encryption_reporting` to determine if there are any existing data containers with unencrypted data.

   More information about how to run the reporting tool is available.

   See "Command usage example outputs" on page 840.

5. If unencrypted data is reported, run the `encryption_reporting` tool again with the `--encrypt` option and wait until it finishes.

   Running the `encryption_reporting` tool with this option, encrypts the identified data containers by the reporting process.

   If the tool with option `--encrypt` reports errors on encrypting the data containers, check the tool logs and MSDP logs for the reasons. When the errors are confirmed, repeat step 4 and step 5 if necessary.

Command usage example outputs

When encryption is not enforced or the rolling data conversion is not finished, the `crcontrol` command denies Encryption Crawler related operations. The following is an example of the output:

```
[root@rsvlmvc01vm0771 /]# /usr/openv/pdde/pdcr/bin/crcontrol --encconvertstate
CRControlEncConvertInfoGet failed : operation not supported
Please double check the server encryption settings
```

Check the data format of a data container before the Encryption Crawler process. The following is an example of the output:

```
[root@rsvlmvc01vm0771 /]# /usr/openv/pdde/pdcr/bin/dcscan --so-data-format 3080|head -n 15
Path = /MSDP/data/3/3080.[bhd, bin]
*** Header for container 3080 ***
version                    : 1
flags                      : 0xe000(DC_ENTRY_FULL|DC_ENTRY_SHA256|DC_ENTRY_BINHEADER)
data file last position    : 67001810
header file last position  : 55252
source id                  : 2505958
retention                  : 0
file size                  : 67001810
delete space               : 0
active records             : 511
total records              : 511
deleted records            : 0
crc32                      : 0x4fd80a49

[root@rsvlmvc01vm0771 /]# /usr/openv/pdde/pdcr/bin/dcscan --so-data-format 3080|tail -n 15
type of record : SO
version        : 4
flags          : 0x2
backup session : 1670238781
fptype         : 3
size           : 131118
record crc     : 4164163489
data crc       : 1313121942
ctime          : 1642086781
offset         : 66870692
digest         : 7f7fd0c5d8fc64d9a7e25c7c079af86613b40d9feff9d316cdfc09c1eafb1690
KMS Enc        : NO
SO crc         : 85135236
data format    : [LZO Compressed Streamable, v2, window size 143360 bytes]

[root@rsvlmvc01vm0771 /]# /usr/openv/pdde/pdcr/bin/dcscan
--so-data-format 3080|grep "data format"|wc
    511    5621   38325
[root@rsvlmvc01vm0771 /]# /usr/openv/pdde/pdcr/bin/dcscan
--so-data-format 3080|grep "data format"|tail -n 5
```

```
data format    : [LZO Compressed Streamable, v2, window size 143360 bytes]
data format    : [LZO Compressed Streamable, v2, window size 143360 bytes]
data format    : [LZO Compressed Streamable, v2, window size 143360 bytes]
data format    : [LZO Compressed Streamable, v2, window size 143360 bytes]
data format    : [LZO Compressed Streamable, v2, window size 143360 bytes]
[root@rsvlmvc01vm0771 /]# /usr/openv/pdde/pdcr/bin/dcscan
--so-data-format 3080|grep "data format"|grep -i -e "AES" -e "Encrypted"
```

Check the data format of a data container after the Encryption Crawler process. The following is an example of the output:

```
[root@rsvlmvc01vm0771 /]# /usr/openv/pdde/pdcr/bin/dcscan --so-data-format 3080|head -n 15
Path = /MSDP/data/3/3080.[bhd, bin]
*** Header for container 3080 ***
version                    : 1
flags                      : 0xe000(DC_ENTRY_FULL|DC_ENTRY_SHA256|DC_ENTRY_BINHEADER)
data file last position    : 67009986
header file last position  : 55252
source id                  : 2505958
retention                  : 0
file size                  : 67009986
delete space               : 0
active records             : 511
total records              : 511
deleted records            : 0
crc32                      : 0x54380a69

[root@rsvlmvc01vm0771 /]# /usr/openv/pdde/pdcr/bin/dcscan --so-data-format 3080|tail -n 15
type of record : SO
version        : 4
flags          : 0x2
backup session : 1670238781
fptype         : 3
size           : 131134
record crc     : 4210300849
data crc       : 1992124019
ctime          : 1642086781
offset         : 66878852
digest         : 7f7fd0c5d8fc64d9a7e25c7c079af86613b40d9feff9d316cdfc09c1eafb1690
KMS Enc        : NO
SO crc         : 85331847
data format    : [AES-256-CTR Encrypted archive 256bit key LZO Compressed Streamable,
v2, window size 143360 bytes]
```

```
[root@rsvlmvc01vm0771 /]# /usr/openv/pdde/pdcr/bin/dcscan
--so-data-format 3080|grep "data format"|wc
    511    8176   59276

[root@rsvlmvc01vm0771 /]# /usr/openv/pdde/pdcr/bin/dcscan
--so-data-format 3080|grep "data format"|tail -n 5
data format    : [AES-256-CTR Encrypted archive 256bit key LZO Compressed Streamable,
v2, window size 143360 bytes]
data format    : [AES-256-CTR Encrypted archive 256bit key LZO Compressed Streamable,
v2, window size 143360 bytes]
data format    : [AES-256-CTR Encrypted archive 256bit key LZO Compressed Streamable,
v2, window size 143360 bytes]
data format    : [AES-256-CTR Encrypted archive 256bit key LZO Compressed Streamable,
v2, window size 143360 bytes]
data format    : [AES-256-CTR Encrypted archive 256bit key LZO Compressed Streamable,
v2, window size 143360 bytes]

[root@rsvlmvc01vm0771 /]# /usr/openv/pdde/pdcr/bin/dcscan
--so-data-format 3080|grep "data format"|grep -i -e "AES" -e "Encrypted"
data format    : [AES-256-CTR Encrypted archive 256bit key LZO Compressed Streamable,
v2, window size 143360 bytes]
data format    : [AES-256-CTR Encrypted archive 256bit key LZO Compressed Streamable,
v2, window size 143360 bytes]
data format    : [AES-256-CTR Encrypted archive 256bit key LZO Compressed Streamable,
v2, window size 143360 bytes]

[root@rsvlmvc01vm0771 /]# /usr/openv/pdde/pdcr/bin/dcscan --so-is-encrypted 3080
1 of 1: unencrypted 0: container 3080: size 67009986
```

Using `dcscan --so-is-encrypted` to check if a container or a list of containers are encrypted.

The status message `unencrypted 0` indicate it's encrypted already, and `unencrypted 1` indicates it's unencrypted and needs to be encrypted. The following is an example of the output:

```
[root@rsvlmvc01vm0771 /]# /usr/openv/pdde/pdcr/bin/dcscan --so-is-encrypted 3080
1 of 1: unencrypted 1: container 3080: size 67001810
```

Using the reporting tool to report on the unencrypted MSDP data.

Cohesity recommends using the reporting tool `encryption_reporting` to report the unencrypted data in the MSDP pool.

Note: The encryption reporting tool is not supported on Flex WORM setups.

Table C-5

OS and Python requirements	Details
Python requirements for `encryption_reporting` on Linux Red Hat installations.	NetBackup Red Hat installations come with Python and there are no extra steps for getting Python running.
Python requirements for `encryption_reporting` on Windows and Linux SUSE BYO installations.	NetBackup 10.0 and newer versions require you to install Python 3.6.8-3.9.16. Currently, no additional software packages are required to be installed. Navigate to the directory containing `encryption_reporting` (`\Veritas\pdde` on windows and `/usr/openv/pdde/pdcr/bin` on Linux SUSE) and run it as a python script.

By default, the reporting tool creates a thread pool of two threads. The tool uses these threads to search for unencrypted data or to encrypt the unencrypted data. A thread is used to process one MSDP mount point to completion. Upon completing the processing of a mount point, the thread is returned to the thread pool. The thread is then used to process any additional mount point that is queued up for processing.

The number of threads is equal to the number of mountpoints that can be processed concurrently. You can increase or decrease the thread pool's thread count by specifying the `-n` option. The minimum thread count is 1 and the maximum is 20.

The reporting tool is I/O intensive. Increasing the thread count up to the total number of MSDP mountpoints usually means better performance for the reporting tool. It also means more load on the system which can affect performance of backup, restore, deduplication, and replication jobs. No performance gains are observed for using more threads than there are mountpoints.

When using the reporting tool to search for the unencrypted data, each thread invokes one instance of `dcscan`. Each `dcscan` instance uses roughly N * 160 MB of memory. In this equation, N is the number of MSDP mountpoints on the server. If there are a total of 12 MSDP mountpoints, each `dcscan` instance uses about 1.8 GB of memory. If there are four threads running in the reporting tool, the reporting tool and the `dcscan` processes consume more than 7 GB of memory.

On a Windows BYO, the default path to `dcscan` is `C:\Program Files\Veritas\pdde`. If you have `dcscan` installed somewhere else, you must use the `-d` or `--dcscan_dir` option to specify the correct location.

The `encryption_reporting` does not account for data encrypted with the Encryption Crawler. If you have previously run the Encryption Crawler to encrypt data, you must clear the metadata files with the `-c` option if they exist. Then re-run `encryption_reporting` to get up-to-date information.

In certain circumstances, data may be reported as `Encrypted needs KMS convert`. This means that the data is encrypted, but not with KMS. If you see this message, use the crawler commands `./crcontrol –encconvertreset` and `./crcontrol –encconverton` to encrypt the rest of the data with KMS.

Cohesity does not recommend that you run the reporting tool while the Encryption Crawler process is active.

Common command lines usage

- `./encryption_reporting –h`
  Display the help output for the command.
- `./encryption_reporting –n 4`
  Reports the amount of unencrypted and encrypted data once the script completes scanning. Use the `-n` option to define the number of threads in the thread pool. The default number of threads is 2.
- `./encryption_reporting –r`
  This command reports the amount of unencrypted data from the metadata files that were generated during a previous scan. It doesn't perform a scan.
- `./encryption_reporting –e –n 4`
  Uses the metadata files to submit data container encryption commands through `crcontrol`. Use the `-n` option to define the number of threads use in the thread pool. The default number of threads is 2.
- `./encryption_reporting –c`
  Delete the metadata files that are created during the scan. Be aware this command deletes all metadata files the previous scan generated.
- `./encryption_reporting`
  Runs the script to determine the amount of encrypted and unencrypted data on the media server.
  This command generates metadata files for each container directory in the MSDP log directory under a directory called `unencrypted_metadata`.
  The script reads in a `configfilepath` from `/etc/pdregistry.cfg` and parses out the path to read in the mount points from `fstab.cfg`. It reads in all mount points in `fstab.cfg`.
  To determine the amount of encrypted and unencrypted data, look for a line similar to the one shown, bold added for emphasis:

```
2021-01-28 17:46:05,555 - root - CRITICAL - unencrypted bytes
58.53GB, encrypted bytes 14.46GB
```

Updating the KMS configuration

For some security or compliance reasons, you may want to make the following changes to the KMS configurations.

- KMS conversion
  Use KMS conversion to convert index-based KMS data containers to KEK-based KMS data containers. You can convert legacy KMS after you upgrade to NetBackup 10.5 from NetBackup 10.2 and earlier versions if KMS encryption was enabled previously. KMS conversion uses Encryption Crawler to update the data containers for datastores with KMS encryption enabled.
  See "Converting the legacy KMS to KEK-based KMS" on page 846.
- KMS key rotation
  Use KMS key rotation to replace the KMS key that is used to encrypt the KEK without changing the underlying existing KEK.
  See "Performing the KMS key rotation" on page 847.
- KMS KEK rotation
  KMS KEK rotation generates a new active KMS KEK and updates the data containers to use the new active KEK's tag. Use KMS KEK rotation if the KEK has been compromised. KMS KEK rotation uses Encryption Crawler to update the data containers for datastores with KMS encryption enabled.
  See "Performing the KEK rotation" on page 847.
- KMS vendor migration
  KMS vendor migration lets you change KMS service providers such as changing from NetBackup KMS to a third-party KMS vendor, a third-party KMS vendor to NetBackup KMS, or third-party KMS vendor A to third-party KMS vendor B. KMS vendor migration requires the user to configure additional KMS service providers before using KMSvendor migration. The KMS key group name on the new KMS service must match the KMS key group name on the current KMS service.
  See "Migrating the KMS vendor" on page 848.

Converting the legacy KMS to KEK-based KMS

NetBackup 10.2 and earlier version use index-based KMS. Data containers written with the legacy index-based KMS can be converted to the current KEK-based KMS using the encryption crawler. This process may run concurrently with the normal NetBackup operations such as backups and restores. Best practice is to perform

KMS conversion during a maintenance window. Conversion may take a very long time to complete since it depends on the amount of data that needs to be converted.

To convert the legacy KMS to KEK-based KMS

1. Reset the encryption crawler if it was used previously.

   ```
   /usr/openv/pdde/pdcr/bin/crcontrol --encconvertreset
   ```

2. Run the following command to start the conversion process in MSDP.

   ```
   /usr/openv/pdde/pdcr/bin/crcontrol --legacykmsconverton
   ```

3. Monitor the conversion progress.

   ```
   /usr/openv/pdde/pdcr/bin/crcontrol --encconvertstate 2
   ```

Performing the KMS key rotation

This process may run concurrently with the normal NetBackup operations such as backups and restores. However, best practice is to perform KMS key rotation during a maintenance window. KMS key rotation should complete quickly.

To perform the KMS key rotation

1. Ensure that you have created a new KMS key on the KMS service you configured MSDP to use.

2. Run the following command to start the KMS key rotation process in MSDP.

   ```
   /usr/openv/pdde/pdcr/bin/crcontrol --kmskeyrotation
   ```

3. Use `kek_tag_reporting` tool to verify that a new KMS key is updated.

   ```
   /usr/openv/pdde/pdcr/bin/kek_tag_reporting.py -r
   ```

   ```
   /usr/openv/netbackup/bin/nbkmscmd -listKeys -name nbkms
   ```

Performing the KEK rotation

This process may run concurrently with normal NetBackup operations such as backups and restores. However, best practice is to perform KEK rotation during a maintenance window. Rotation may take a very long time to complete since it depends on the amount of data that needs KEK rotation.

To perform the KEK rotation

1. Reset the encryption crawler if it was used previously.

   ```
   /usr/openv/pdde/pdcr/bin/crcontrol --encconvertreset
   ```

2. Run the following command to start the KEK rotation process in MSDP.

   ```
   /usr/openv/pdde/pdcr/bin/crcontrol --kekconverton
   ```

3. Monitor the KEK rotation.

   ```
   /usr/openv/pdde/pdcr/bin/crcontrol --encconvertstate 2
   ```

Migrating the KMS vendor

Best practice is to perform KMS vendor migration during a maintenance window.

To migrate the KMS vendor

1. List the current KMS service to determine the KMS service priority.

   ```
   /usr/openv/netbackup/bin/nbkmscmd -listKMSConfig
   ```

2. Update the priority on the current KMS service to greater than 0.

   ```
   /usr/openv/netbackup/bin/nbkmscmd -updateKMSConfig -name
   configuration_name [-server primary_server_name] [-priority
   priority_of_KMS_server]
   ```

3. Set up the new KMS service with the same key group name the current KMS service is using.
4. Create an active KMS key in the new KMS service.
5. Configure new KMS service in NetBackup with priority of 0.
6. Verify that NetBackup reports both KMS services on the primary server.

   ```
   /usr/openv/netbackup/bin/nbkmscmd -listKMSConfig
   ```

7. Update the priority of the new KMS service to a priority greater than the priority that is set on the previous KMS service.

   ```
   /usr/openv/netbackup/bin/nbkmscmd -updateKMSConfig -name
   configuration_name [-server primary_server_name] [-priority
   priority_of_KMS_server]
   ```

8 Start the KMS vendor migration process in MSDP.

```
/usr/openv/pdde/pdcr/bin/crcontrol --migratekmsprovider
```

9 Use `kek_tag_reporting` tool to verify the `kms_key_tag` in the most recent entry matches the "Key ID" reported by `nbkmscmd` for active key in the new KMS service.

```
/usr/openv/pdde/pdcr/bin/kek_tag_reporting.py -r
```

```
/usr/openv/netbackup/bin/nbkmscmd -listKeys -name nbkms
```

Index

Symbols

521

A

AES encryption
    Blowfish encryption 113
AES-256 111
appliance deduplication 23
attributes
    clearing deduplication pool 548
    clearing deduplication storage server 534
    setting deduplication pool 543
    setting deduplication storage server 533
    viewing deduplication pool 542
    viewing deduplication storage server 532
Auto Image Replication
    Backup operation in source domain 137
    configuring MSDP replication to a different domain 137

B

backup
    client deduplication process 598

C

capacity and usage reporting for deduplication 521
changing deduplication server hostname 535
clearing deduplication pool attributes 548
client deduplication
    components 597
compacting container files 523
compression
    for MSDP backups 108
    for MSDP optimized duplication and replication 109
    pd.conf file setting 180
configuring deduplication 65
container files
    about 523
    compaction 523
container files *(continued)*
    viewing capacity within 523
contentrouter.cfg file
    about 194
    parameters for data integrity checking 559
    ServerOptions for encryption 110
crcontrol 112
credentials 42
    adding NetBackup Deduplication Engine 540
    changing NetBackup Deduplication Engine 541

D

data conversion
    encryption 111
data integrity checking
    about deduplication 554
    configuration settings for deduplication 557
    configuring behavior for deduplication 555
data removal process
    for deduplication 563
database system error 774
deactivating media server deduplication 591
deduplication
    about credentials 42
    adding credentials 540
    and Fibre Channel 32
    and iSCSI 32
    capacity and usage reporting 521
    changing credentials 541
    client backup process 598
    configuration file 178
    configuring 65
    configuring optimized synthetic backups 120
    container files 523
    data removal process 563
    limitations 38
    media server process 596
    network interface 43
    node 27
    scaling 51

deduplication configuration file
editing 72, 179
parameters 180
deduplication data integrity checking
about 554
configuring behavior for 555
settings 557
deduplication deduplication pool. *See* deduplication pool
deduplication disk volume
changing the state 550
determining the state 549
deduplication encryption
enabling for MSDP backups 109
deduplication host configuration file 199
deleting 200
deduplication hosts
and firewalls 43
load balancing server 36
deduplication logs
client deduplication proxy plug-in log 769
deduplication node
about 27
deduplication plug-in
about 594
deduplication plug-in configuration file
configuring 67
deduplication pool. *See* deduplication pool
about 97
clearing attributes 548
properties 100
setting attributes 543
viewing attributes 542
deduplication port usage
about 43
troubleshooting 785
deduplication processes do not start 776
deduplication reference database
about 594
deduplication registry
resetting 200
deduplication servers
components 593
deduplication storage capacity
viewing capacity in container files 523
deduplication storage server
clearing attributes 534
components 593
deleting the configuration 539
deduplication storage server *(continued)*
determining the state 531
editing configuration file 197
getting the configuration 196
setting attributes 533
setting the configuration 198
viewing attributes 532
deduplication storage server configuration file
about 195
deduplication storage server name
changing 535
deleting backup images 552
deleting deduplication host configuration file 200
df command 786
disaster recovery
protecting the data 55
disk failure
deduplication storage server 583
disk logs report 523
disk pool status report 522
disk pools
cannot delete 781
disk volume
changing the state 550
determining the state of a deduplication 549
volume state changes to down 779
duplication jobs, cancelling 782

E

Enable file recovery from VM backup 169
encryption
enabling for MSDP backups 109
pd.conf file setting 183
SHA-2 111

F

Fibre Channel
and iSCSI comparison 32
Fibre Channel and iSCSI comparison for MSDP 32
firewalls and deduplication hosts 43

G

garbage collection. *See* queue processing

I

iSCSI
and Fibre Channel comparison 32

J

job ID search in unified logs 765

L

limitations
    media server deduplication 38
load balancing server
    about 36
    for deduplication 36
logs
    client deduplication proxy plug-in log 769

M

maintenance processing. *See* queue processing
media server deduplication
    process 596
Media Server Deduplication Pool 97
    creating directories for 400 TB support 89
    enable 400 TB support 89
media server deduplication pool. *See* deduplication pool
migrating to NetBackup deduplication 797
MSDP
    replication 136
MSDP catalog 201
    See also MSDP catalog backup
    about the catalog backup policy 202
    about the shadow catalog 201
    changing the number of shadow copies 206
    changing the shadow catalog path 204
    changing the shadow catalog schedule 205
    shadow copy log files 768
MSDP catalog backup
    about protecting the MSDP catalog 202
    configuring 211
MSDP catalog recovery
    process the transation queue. 553
MSDP drcontrol utility
    options 208
MSDP replication
    about 45
MSDP storage rebasing. *See* rebasing
mtstrm.conf file
    configuring 67

N

NetBackup appliance deduplication 23
NetBackup deduplication
    about 22
NetBackup Deduplication Engine
    about 594
    about credentials 42
    adding credentials 540
    changing credentials 541
NetBackup Deduplication Manager
    about 594
NetBackup deduplication options 22
network interface
    for deduplication 43
node
    deduplication 27

O

optimized deduplication
    configuring bandwidth 160
    configuring for MSDP 129
    separate network for 121
optimized deduplication copy
    guidance for 125
optimized duplication
    about 45
    about the media server in common within the same domain 125
optimized synthetic backups
    configuring for deduplication 120

P

pd.conf file
    about 178
    editing 72, 179
    parameters 180
port usage
    and deduplication 43
    troubleshooting 785
PureDisk Deduplication Pool 97

Q

queue processing 552
    invoke manually 553

R

rebasing
    about 561
    FP_CACHE_PERIOD_REBASING_THRESHOLD parameter 186

rebasing *(continued)*
FP_CACHE_REBASING_THRESHOLD parameter 186
RebaseMaxPercentage parameter 563
RebaseMaxTime parameter 563
RebaseMinContainers parameter 563
RebaseScatterThreshold parameter 563
server-side rebasing parameters 562
recovery
from deduplication storage server disk failure 583
Red Hat Linux
deduplication processes do not start 776
replication
configuring MSDP replication to a different domain 137
for MSDP 45, 136
reports
disk logs 523
disk pool status 522
resetting the deduplication registry 200
restores
how deduplication restores work 565

S

scaling deduplication 51
server not found error 774
setting deduplication pool attributes 543
SHA-2 111
SHA-512/256 111
spa.cfg file
parameters for data integrity checking 559
storage capacity
viewing capacity in container files 523
storage lifecycle policies
cancelling duplication jobs 782
storage paths
about reconfiguring 535
storage rebasing. *See* rebasing
storage server
about the configuration file 195
changing the name 535
components for deduplication 593
deleting the deduplication configuration 539
determining the state of a deduplication 531
editing deduplication configuration file 197
getting deduplication configuration 196
setting the deduplication configuration 198
viewing attributes 532
storage server configuration
getting 196
setting 198
storage server configuration file
editing 197
storage unit
recommendations for deduplication 104
storage unit groups
not supported for Auto Image Replication source 137
stream handlers
NetBackup 46

T

troubleshooting
database system error 774
deduplication backup jobs fail 776
deduplication processes do not start 776
general operational problems 780
server not found error 774

U

unified logging 761
format of files 762
uninstalling media server deduplication 591

V

viewing deduplication pool attributes 542
viewing storage server attributes 532
VM backup 169
vxlogview command 762
with job ID option 765