NetBackup™ Deduplication Guide

UNIX, Windows, and Linux

Release 10.1

VERITAS™

NetBackup™ Deduplication Guide

Last updated: 2022-09-07

Legal Notice

Copyright © 2022 Veritas Technologies LLC. All rights reserved.

Veritas, the Veritas Logo, and NetBackup are trademarks or registered trademarks of Veritas Technologies LLC or its affiliates in the U.S. and other countries. Other names may be trademarks of their respective owners.

This product may contain third-party software for which Veritas is required to provide attribution to the third party (“Third-party Programs”). Some of the Third-party Programs are available under open source or free software licenses. The License Agreement accompanying the Software does not alter any rights or obligations you may have under those open source or free software licenses. Refer to the Third-party Legal Notices document accompanying this Veritas product or available at:

https://www.veritas.com/about/legal/license-agreements

The product described in this document is distributed under licenses restricting its use, copying, distribution, and decompilation/reverse engineering. No part of this document may be reproduced in any form by any means without prior written authorization of Veritas Technologies LLC and its licensors, if any.

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. Veritas Technologies LLC 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.

The Licensed Software and Documentation are deemed to be commercial computer software as defined in FAR 12.212 and subject to restricted rights as defined in FAR Section 52.227-19 "Commercial Computer Software - Restricted Rights" and DFARS 227.7202, et seq. "Commercial Computer Software and Commercial Computer Software Documentation," as applicable, and any successor regulations, whether delivered by Veritas as on premises or hosted services. Any use, modification, reproduction release, performance, display or disclosure of the Licensed Software and Documentation by the U.S. Government shall be solely in accordance with the terms of this Agreement.

Veritas Technologies LLC
2625 Augustine Drive
Santa Clara, CA 95054

http://www.veritas.com

Technical Support

Technical Support maintains support centers globally. All support services will be delivered in accordance with your support agreement and the then-current enterprise technical support policies. For information about our support offerings and how to contact Technical Support, visit our website:

https://www.veritas.com/support

You can manage your Veritas account information at the following URL:

https://my.veritas.com

If you have questions regarding an existing support agreement, please email the support agreement administration team for your region as follows:

Worldwide (except Japan)	CustomerCare@veritas.com
Japan	CustomerCare_Japan@veritas.com

Documentation

Make sure that you have the current version of the documentation. Each document displays the date of the last update on page 2. The latest documentation is available on the Veritas website:

https://sort.veritas.com/documents

Documentation feedback

Your feedback is important to us. Suggest improvements or report errors or omissions to the documentation. Include the document title, document version, chapter title, and section title of the text on which you are reporting. Send feedback to:

NB.docs@veritas.com

You can also see documentation information or ask a question on the Veritas community site:

http://www.veritas.com/community/

Veritas Services and Operations Readiness Tools (SORT)

Veritas Services and Operations Readiness Tools (SORT) is a website that provides information and tools to automate and simplify certain time-consuming administrative tasks. Depending on the product, SORT helps you prepare for installations and upgrades, identify risks in your datacenters, and improve operational efficiency. To see what services and tools SORT provides for your product, see the data sheet:

https://sort.veritas.com/data/support/SORT_Data_Sheet.pdf

Contents

Chapter 1 Introducing the NetBackup media server deduplication option 16

About the NetBackup deduplication options 16

Chapter 2 Quick start 18

About client-side deduplication 18
About the media server deduplication (MSDP) node cloud tier 20
Configuring the MSDP node cloud tier 21
About Auto Image Replication (A.I.R.) 23

Chapter 3 Planning your deployment 29

Planning your MSDP deployment 30
NetBackup naming conventions 31
About MSDP deduplication nodes 32
About the NetBackup deduplication destinations 32
About MSDP storage capacity 33
About MSDP storage and connectivity requirements 34
Fibre Channel and iSCSI comparison for MSDP 36
About NetBackup media server deduplication 37
About MSDP storage servers 39
About MSDP load balancing servers 40
About MSDP server requirements 40
About MSDP unsupported configurations 42
About NetBackup Client Direct deduplication 42
About MSDP client deduplication requirements and limitations 44
About MSDP remote office client deduplication 44
About MSDP remote client data security 45
About remote client backup scheduling 45
About the NetBackup Deduplication Engine credentials 45
About the network interface for MSDP 46
About MSDP port usage 47
About MSDP optimized synthetic backups 48
About MSDP and SAN Client 48

About MSDP optimized duplication and replication 49
About MSDP performance 49
How file size may affect the MSDP deduplication rate 50
About MSDP stream handlers 50
Oracle stream handler 51
Microsoft SQL Server stream handler 53
MSDP deployment best practices 54
Use fully qualified domain names 54
About scaling MSDP 55
Send initial full backups to the storage server 55
Increase the number of MSDP jobs gradually 56
Introduce MSDP load balancing servers gradually 56
Implement MSDP client deduplication gradually 57
Use MSDP compression and encryption 57
About the optimal number of backup streams for MSDP 57
About storage unit groups for MSDP 58
About protecting the MSDP data 58
Save the MSDP storage server configuration 59
Plan for disk write caching 59

Chapter 4 Provisioning the storage 60

About provisioning the storage for MSDP 60
Do not modify MSDP storage directories and files 62
About volume management for NetBackup MSDP 62

Chapter 5 Licensing deduplication 64

About the MSDP license 64
Licensing NetBackup MSDP 65

Chapter 6 Configuring deduplication 66

Configuring MSDP server-side deduplication 69
Configuring MSDP client-side deduplication 71
About the MSDP Deduplication Multi-Threaded Agent 72
Configuring the Deduplication Multi-Threaded Agent behavior 74
MSDP mtstrm.conf file parameters 75
Configuring deduplication plug-in interaction with the Multi-Threaded Agent 79
About MSDP fingerprinting 80
About the MSDP fingerprint cache 81
Configuring the MSDP fingerprint cache behavior 82
MSDP fingerprint cache behavior options 82

About seeding the MSDP fingerprint cache for remote client deduplication 83
Configuring MSDP fingerprint cache seeding on the client 85
Configuring MSDP fingerprint cache seeding on the storage server 86
NetBackup seedutil options 88
About sampling and predictive cache 89
Enabling 400 TB support for MSDP 90
About MSDP Encryption using NetBackup KMS service 91
Upgrading KMS for MSDP 92
Enabled KMS encryption for Local LSU 94
About MSDP Encryption using external KMS server 95
Configuring a storage server for a Media Server Deduplication Pool 95
MSDP storage path properties 110
MSDP network interface properties 113
About disk pools for NetBackup deduplication 113
Configuring a disk pool for deduplication 114
Media Server Deduplication Pool properties 116
Creating the data directories for 400 TB MSDP support 118
Adding volumes to a 400 TB Media Server Deduplication Pool 118
Configuring a Media Server Deduplication Pool storage unit 121
Media Server Deduplication Pool storage unit properties 122
MSDP storage unit recommendations 124
Configuring client attributes for MSDP client-side deduplication 125
Disabling MSDP client-side deduplication for a client 127
About MSDP compression 127
About MSDP encryption 129
MSDP compression and encryption settings matrix 130
Configuring encryption for MSDP backups 132
Configuring encryption for MSDP optimized duplication and replication 134
About the rolling data conversion mechanism for MSDP 135
Modes of rolling data conversion 135
MSDP encryption behavior and compatibilities 138
Configuring optimized synthetic backups for MSDP 139
About a separate network path for MSDP duplication and replication 139
Configuring a separate network path for MSDP duplication and replication 140
About MSDP optimized duplication within the same domain 141
About the media servers for MSDP optimized duplication within the same domain 143

Configuring MSDP optimized duplication within the same NetBackup domain 148
Configuring NetBackup optimized duplication or replication behavior 152
About MSDP replication to a different domain 155
Configuring MSDP replication to a different NetBackup domain 156
About NetBackup Auto Image Replication 158
About trusted primary servers for Auto Image Replication 165
About the certificate to be used for adding a trusted master server 169
Adding a trusted master server using a NetBackup CA-signed (host ID-based) certificate 171
Adding a trusted primary server using external CA-signed certificate 176
Removing a trusted primary server 177
Enabling NetBackup clustered primary server inter-node authentication 179
Configuring NetBackup CA and NetBackup host ID-based certificate for secure communication between the source and the target MSDP storage servers 180
Configuring external CA for secure communication between the source MSDP storage server and the target MSDP storage server 181
Configuring a target for MSDP replication to a remote domain 182
About configuring MSDP optimized duplication and replication bandwidth 188
About performance tuning of optimized duplication and replication for MSDP cloud 189
About storage lifecycle policies 189
About the storage lifecycle policies required for Auto Image Replication 190
Creating a storage lifecycle policy 192
Storage Lifecycle Policy dialog box settings 195
About MSDP backup policy configuration 198
Creating a backup policy 199
Resilient Network properties 199
Resilient connection resource usage 202
Specifying resilient connections 203
Adding an MSDP load balancing server 205
About variable-length deduplication on NetBackup clients 206
About the MSDP pd.conf configuration file 208
Editing the MSDP pd.conf file 209

MSDP pd.conf file parameters 209
About the MSDP contentrouter.cfg file 224
About saving the MSDP storage server configuration 225
Saving the MSDP storage server configuration 226
Editing an MSDP storage server configuration file 227
Setting the MSDP storage server configuration 228
About the MSDP host configuration file 229
Deleting an MSDP host configuration file 229
Resetting the MSDP registry 230
About protecting the MSDP catalog 231
About the MSDP shadow catalog 231
About storing MSDP catalog shadow copy duplicates on data volumes 232
About the MSDP catalog backup policy 232
Changing the MSDP shadow catalog path 234
Changing the MSDP shadow catalog schedule 235
Changing the number of MSDP catalog shadow copies 236
Configuring an MSDP catalog backup 237
MSDP drcontrol options 238
Updating an MSDP catalog backup policy 241
About MSDP FIPS compliance 243
Configuring the NetBackup client-side deduplication to support multiple interfaces of MSDP 245
About MSDP multi-domain support 245
About MSDP application user support 249
About MSDP mutli-domain VLAN Support 250
About NetBackup WORM storage support for immutable and indelible data 251
About the NetBackup command line options to configure immutable and indelible data 252

Chapter 7 MSDP cloud support 255

About MSDP cloud support 256
Operating system requirement for configuration 257
Limitations 257
Create a Media Server Deduplication Pool (MSDP) storage server in the NetBackup web UI 258
Creating a cloud storage unit 259
Updating cloud credentials for a cloud LSU 263
Updating encryption configurations for a cloud LSU 264
Deleting a cloud LSU 265
Backup data to cloud by using cloud LSU 267

Duplicate data cloud by using cloud LSU 267
Configuring AIR to use cloud LSU 267
About backward compatibility support 271
About the configuration items in cloud.json, contentrouter.cfg, and spa.cfg 272
Cloud space reclamation 277
Configuring the container aging 277
Configuring the cloud compaction 278
About the tool updates for cloud support 279
About the disaster recovery for cloud LSU 281
Common disaster recovery steps 284
Disaster recovery for cloud LSU in Flex Scale 289
About Image Sharing using MSDP cloud 290
Things to consider before you use image sharing to convert VM image to VHD in Azure 300
Converting the VM image to VHD in Azure 302
About restore from a backup in Microsoft Azure Archive 306
About MSDP cloud immutable (WORM) storage support 307
About MSDP cloud admin tool 307
About immutable object support for AWS S3 308
About immutable object support for AWS S3 compatible platforms 315
About immutable storage support for Azure blob storage 325
About using the cloud immutable storage in a cluster environment 329
Troubleshooting the error when the bucket is created without msdpcldutil 330
Deleting the immutable image with the governance mode 331
Refreshing the immutable cloud volume retention range value in the web UI 331
Deleting the S3 object permanently 331
About instant access for object storage in cloud 332

Chapter 8 S3 Interface for MSDP 333

About S3 interface for MSDP 333
Prerequisites 334
Configuring S3 interface for MSDP 334
Changing the certificate in S3 server 336
Identity and Access Management (IAM) for S3 interface for MSDP 336
Signing IAM and S3 API requests 336
IAM workflow 336

IAM APIs for S3 interface for MSDP 339
IAM policy document syntax 362
S3 APIs for S3 interface for MSDP 364
S3 APIs on Buckets 365
S3 APIs on Objects 388
The naming rules for buckets and objects 405
Disaster recovery in S3 interface for MSDP 406
Limitations in S3 interface for MSDP 407
Logging and troubleshooting 407
Best practices 408

Chapter 9 Monitoring deduplication activity 409

Monitoring the MSDP deduplication and compression rates 409
Viewing MSDP job details 411
MSDP job details 412
About MSDP storage capacity and usage reporting 414
About MSDP container files 416
Viewing storage usage within MSDP container files 416
Viewing MSDP disk reports 417
About monitoring MSDP processes 419
Reporting on Auto Image Replication jobs 419

Chapter 10 Managing deduplication 421

Managing MSDP servers 422
Viewing MSDP storage servers 422
Determining the MSDP storage server state 422
Viewing MSDP storage server attributes 423
Setting MSDP storage server attributes 424
Changing MSDP storage server properties 425
Clearing MSDP storage server attributes 426
About changing the MSDP storage server name or storage path 427
Changing the MSDP storage server name or storage path 427
Removing an MSDP load balancing server 429
Deleting an MSDP storage server 430
Deleting the MSDP storage server configuration 431
Managing NetBackup Deduplication Engine credentials 432
Determining which media servers have deduplication credentials 432
Adding NetBackup Deduplication Engine credentials 432
Changing NetBackup Deduplication Engine credentials 433
Deleting credentials from a load balancing server 433

Managing Media Server Deduplication Pools 434
Viewing Media Server Deduplication Pools 434
Determining the Media Server Deduplication Pool state 434
Changing OpenStorage disk pool state 435
Viewing Media Server Deduplication Pool attributes 435
Setting a Media Server Deduplication Pool attribute 436
Changing a Media Server Deduplication Pool properties 437
Clearing a Media Server Deduplication Pool attribute 442
Determining the MSDP disk volume state 443
Changing the MSDP disk volume state 444
Inventorying a NetBackup disk pool 444
Deleting a Media Server Deduplication Pool 445
Deleting backup images 445
About MSDP queue processing 446
Processing the MSDP transaction queue manually 446
About MSDP data integrity checking 448
Configuring MSDP data integrity checking behavior 448
MSDP data integrity checking configuration parameters 451
About managing MSDP storage read performance 452
About MSDP storage rebasing 453
MSDP server-side rebasing parameters 455
About the MSDP data removal process 455
Resizing the MSDP storage partition 456
How MSDP restores work 457
Configuring MSDP restores directly to a client 458
About restoring files at a remote site 459
About restoring from a backup at a target master domain 459
Specifying the restore server 460

Chapter 11 Recovering MSDP 462

About recovering the MSDP catalog 462
Restoring the MSDP catalog from a shadow copy 463
Recovering from an MSDP storage server disk failure 465
Recovering from an MSDP storage server failure 466
Recovering the MSDP storage server after NetBackup catalog recovery 469

Chapter 12 Replacing MSDP hosts 470

Replacing the MSDP storage server host computer 470

Chapter 13 Uninstalling MSDP 473

About uninstalling MSDP 473
Deactivating MSDP 473

Chapter 14 Deduplication architecture 475

MSDP server components 475
Media server deduplication backup process 478
MSDP client components 479
MSDP client–side deduplication backup process 480

Chapter 15 Configuring and using universal shares 483

About universal shares 484
Configuring and using an MSDP build-your-own (BYO) server for universal shares 486
MSDP build-your-own (BYO) server prerequisites and hardware requirements to configure universal shares 489
Configuring universal share user authentication 491
Mounting a universal share created from the NetBackup web UI 495
Creating a protection point for a universal share 497
About universal share self-service recovery 499
Performing a universal share self-service recovery 499
Using the ingest mode 500
Using the ingest mode to take a snapshot over NFS or SMB 501
Using the ingest mode to run a policy using NFS or SMB 501
About universal shares with object store 502
Enabling a universal share with object store 503
Enabling instant access with object storage 506
Disaster recovery for a universal share 506
Changing the number of vpfsd instances 508
Upgrading to NetBackup 10.1 509

Chapter 16 Configuring isolated recovery environment (IRE) 510

Configuring an isolated recovery environment on a NetBackup BYO media server 510
Managing an isolated recovery environment on a NetBackup BYO media server 515
Configuring AIR for replicating backup images from production environment to IRE BYO environment 518

Chapter 17 Using the NetBackup Deduplication Shell 522

About the NetBackup Deduplication Shell 522
Managing users from the deduplication shell 523
Adding and removing local users from the deduplication shell 524
Adding MSDP users from the deduplication shell 525
Connecting an Active Directory domain to a WORM or an MSDP storage server for Universal Shares and Instant Access 526
Disconnecting an Active Directory domain from the deduplication shell 527
Changing a user password from the deduplication shell 527
Managing VLAN interfaces from the deduplication shell 529
Managing the retention policy on a WORM storage server 529
Managing images with a retention lock on a WORM storage server 530
Auditing WORM retention changes 531
Managing certificates from the deduplication shell 532
Viewing the certificate details from the deduplication shell 532
Importing certificates from the deduplication shell 533
Removing certificates from the deduplication shell 535
Managing FIPS mode from the deduplication shell 536
Encrypting backups from the deduplication shell 536
Tuning the MSDP configuration from the deduplication shell 537
Setting the MSDP log level from the deduplication shell 539
Managing NetBackup services from the deduplication shell 540
Managing the cyclic redundancy checking (CRC) service 541
Managing the content router queue processing (CRQP) service 542
Managing the online checking service 543
Managing the compaction service 543
Managing the deduplication (MSDP) services 544
Managing the Storage Platform Web Service (SPWS) 545
Managing the Veritas provisioning file system (VPFS) mounts 545
Managing the NGINX service 546
Managing the SMB service 547
Monitoring and troubleshooting NetBackup services from the deduplication shell 547
About the support command 547

Chapter 18 Troubleshooting 561

About unified logging 561
About using the vxlogview command to view unified logs 562
Examples of using vxlogview to view unified logs 564
About legacy logging 565
Creating NetBackup log file directories for MSDP 566
NetBackup MSDP log files 567
Troubleshooting MSDP installation issues 573
MSDP installation on SUSE Linux fails 573
Troubleshooting MSDP configuration issues 574
MSDP storage server configuration fails 574
MSDP database system error (220) 574
MSDP server not found error 575
License information failure during MSDP configuration 575
The disk pool wizard does not display an MSDP volume 576
Troubleshooting MSDP operational issues 576
Verify that the MSDP server has sufficient memory 577
MSDP backup or duplication job fails 577
MSDP client deduplication fails 579
MSDP volume state changes to DOWN when volume is unmounted 580
MSDP errors, delayed response, hangs 581
Cannot delete an MSDP disk pool 581
MSDP media open error (83) 582
MSDP media write error (84) 584
MSDP no images successfully processed (191) 586
MSDP storage full conditions 586
Troubleshooting MSDP catalog backup 587
Storage Platform Web Service (spws) does not start 588
Disk volume API or command line option does not work 588
Viewing MSDP disk errors and events 588
MSDP event codes and messages 588
Unable to obtain the administrator password to use an AWS EC2 instance that has a Windows OS 591
Trouble shooting multi-domain issues 592
Unable to configure OpenStorage server from another domain 592
MSDP storage server is down when you configure an OpenStorage server 592
MSDP server is overloaded when it is used by multiple NetBackup domains 593
Troubleshooting the cloud compaction error messages 593

Appendix A Migrating to MSDP storage 595

Migrating from another storage type to MSDP 595

Appendix B Migrating from Cloud Catalyst to MSDP direct cloud tiering 597

About migration from Cloud Catalyst to MSDP direct cloud tiering 597
About Cloud Catalyst migration strategies 598
About direct migration from Cloud Catalyst to MSDP direct cloud tiering 603
About requirements for a new MSDP direct cloud tier storage server 603
About beginning the direct migration 604
Placing the Cloud Catalyst server in a consistent state 605
About installing and configuring the new MSDP direct cloud tier server 607
Running the migration to the new MSDP direct cloud tier server 609
About postmigration configuration and cleanup 614
About the Cloud Catalyst migration `-dryrun` option 616
About Cloud Catalyst migration `cacontrol` options 617
Reverting back to Cloud Catalyst from a successful migration 619
Reverting back to Cloud Catalyst from a failed migration 622

Appendix C Encryption Crawler 625

About the Encryption Crawler 625
About the two modes of the Encryption Crawler 626
Managing the Encryption Crawler 628
Advanced options 633
Tuning options 634
Encrypting the data 637
Command usage example outputs 638

Index 645

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

Veritas 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 37.
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 42.
NetBackup appliance deduplication	Veritas provides several hardware and a software solutions that include NetBackup deduplication. The NetBackup appliances have their own documentation set: https://www.veritas.com/content/support/en_US/Appliances.html

Chapter 2

Quick start

This chapter includes the following topics:

- About client-side deduplication
- About the media server deduplication (MSDP) node cloud tier
- About Auto Image Replication (A.I.R.)

About client-side deduplication

Client-side deduplication or Client Direct is an easy way to improve the performance of your backups to an MSDP target. Part of the innovated MSDP deduplication architecture is the use of a distributed, plugin-based fingerprinting service. Instead moving all the data to the storage server before it's deduplicated, the fingerprinting, compression, and encryption can all be performed right on the source. This leads to ideal optimization and acceleration, with minimal network overhead. In the past, with lower power CPUs compared to today's technology, Client Direct was only recommended for systems with high-power processors. Testing has shown that the effect to a client system is very low. As a result, the use of client-side deduplication is encouraged for wider, more regular use.

The three **Deduplication Location** options for MSDP are:

- **Always use the media server** - All data is sent to the media server and the plug-in deduplication occurs on that server before the MSDP storage target is written to.
- **Prefer to use client-side deduplication** – At the beginning of a backup, a quick test is performed to verify that the client can successfully use client-side deduplication. If the test fails, the job falls back on the use of server-side deduplication.
- **Always use client-side deduplication** – The backup job explicitly uses client-side deduplication. If the functionality does not work, the job fails.

Note: When deduplication is performed on the server side or the client side, the same plug-in library is loaded. As a result, the deduplication capabilities and results are not different.

How to enable client-side deduplication

By default, deduplication from the client side is disabled, and must be enabled on a per host basis. From a policy perspective, the functionality can be explicitly disabled. If you include the command line, there are three ways to control this setting.

The three ways to control the setting are as follows:

1. To enable client-side deduplication, you must add the client to the `clientDB` and then setting the client to **Prefer to use client-side deduplication**.
   - To do this operation in the Java GUI, first open the master server host properties, and then open **Client attributes** section.
   - Select **Prefer to use client-side deduplication** from the **Deduplication location** drop-down and select **OK**.
2. To enable client-side deduplication the command line, use with the `bpclient` command with the `-client_direct` option. Refer to the following example for `-client_direct` usage:

   ```
   -client_direct <0=Deduplicate on the media server or
   Move data via media server,
   1=Prefer to use client-side deduplication or
   Prefer to move data direct to storage,
   2=Always use client-side deduplication or
   Always move data direct to storage>
   ```

   The following is an example of how to use the `bpblient` command with the `-client_option` to add the client to the `clientDB` and enable **Prefer to use client-sided deduplication**:

   - UNIX:

     ```
     /usr/openv/NetBackup/bin/admincmd/bpclient
     -client <CLIENT_NAME> -add -client_direct 1
     ```

   - Windows:

     ```
     \Program Files\Veritas\NetBackup\bin\admincmd\bpclient.exe
      -client <CLIENT_NAME> -add -client_direct 1
     ```

3. You can use a script to enable client-side deduplication. The following is an example of a script that checks if the client exists and if not, it adds the client and enables **Prefer to use client-sided deduplication**. If the client already exists, the script updates the setting to **Prefer to use client-sided deduplication**.

   Script example:

```
> export CLIENTLIST = ”client1 client2 client3 client4”
#!/bin/bash
for CLIENT in $CLIENTLIST
do
/usr/openv/NetBackup/bin/admincmd/bpclient
-client $CLIENT -l &> /dev/null
EXISTS=$?
if [ $EXISTS = "227" ]
then
echo "$CLIENT not found, adding and enabling client direct"
/usr/openv/NetBackup/bin/admincmd/bpclient
-client  $CLIENT -add -client_direct 1 ;
else
echo "Updating $CLIENT to use client direct"
/usr/openv/NetBackup/bin/admincmd/bpclient
-client  $CLIENT -update -client_direct 1 ;
fi;
done
```

Note: To disable the use of client-side deduplication on a per policy basis, you must select **Disable client-side deduplication** for each policy in the **Attributes** tab.

About the media server deduplication (MSDP) node cloud tier

Starting with NetBackup 8.3, an MSDP server is able to directly write deduplicated data to cloud object storage. The cloud-tiering feature automatically uses the local block storage pool as its write-cache. This setup creates performance and efficiency improvements and prevents a network hop or requiring a dedicated cache when the cloud object storage is written to. To simplify deployment, MSDP cloud tiering enables data management in multiple buckets, storage tiers, and cloud providers from a single node.

Some of the key attributes of the MSDP cloud-tiering feature include:

- Fewer servers required
- Increased performance
- Multi-bucket support
- Easy web UI configuration
- API-based deployment
- Self-descriptive storage

Requirements for MSDP cloud tier:

- **Hardware requirements for block storage only MSDP pool** - No change from NetBackup 8.2 MSDP guidance. Max capacity is 960 TB for the NetBackup Appliance, and 400 TB for BYO MSDP.
- **Hardware requirements for object storage only pool** - Max capacity of 2 PB and 196 GB of memory. The default is 1 TB of local storage per cloud LSU, and the overall file system utilization should not exceed 90% full.
- **Hardware requirements for mixed object and block storage** - Similar hardware requirements as local storage only pool. Total max capacity is 2.4 PB.
- **Operating system** - Cloud Logical storage units (LSUs) can be configured on the storage servers running on Red Hat Linux Enterprise or CentOS platforms. No platform limitations for clients and load-balancing servers.

Features of the MSDP cloud tier:

- 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 to multiple cloud targets simultaneously.
- The cloud targets can be from the same or from different providers, either public, or private. For example, AWS, Azure, and HCP. These 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 from different cloud providers.
- Based on the OpenStorage Technology (OST), the new architecture uses multiple LSUs 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.

Configuring the MSDP node cloud tier

After you upgrade or install NetBackup 8.3 or later and configure MSDP, cloud tiering can be done by performing the following procedure in the web UI.

To configure the MSDP node cloud tier

1. On the left, click **Storage**, click the **Disk pools** tab, and then 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. In **Volumes**, use the **Volume** drop down to select a volume or add a new volume. Provide a unique volume name that gives adequate description of the volume.

   In the **Cloud storage provider** section, select the cloud provider name from the drop-down list.

   In the **Region** section, select the appropriate region.

   Enter the credentials to complete the setup. You can configure additional options here such as adding a proxy server.

   In the **Select cloud bucket** section, you can create a cloud bucket by clicking **Add** or select a predefined bucket from the list. If the cloud credentials in use do not have the permissions to list buckets, then manually enter a predefined 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. Click **Add storage unit** at the top of the screen.
7. Select **Media Server Deduplication Pool (MSDP)** from the list and click **Start**.
8. In **Basic properties**, enter the **Name** of the MSDP storage unit and click **Next**.
9. In **Disk pool**, select the disk pool that was created and then click **Next**.

10 In the **Media server** tab, use the default selection of **Allow NetBackup to automatically select**and then click **Next**.

11 Review the setup of the storage unit and then click **Save**.

About Auto Image Replication (A.I.R.)

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 (A.I.R.).

Table 2-1 Supported A.I.R. models

Model	Description
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.
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 the following storage types for A.I.R.:

- Media Server Deduplication Pool (MSDP)
- An OpenStorage disk appliance that supports replication

NetBackup uses storage lifecycle policies (SLP) in the source domain and the target domain to manage A.I.R. operations. The following table is a process overview of A.I.R., generally describing the events in the originating and target domains.

Table 2-2 Process overview of A.I.R.

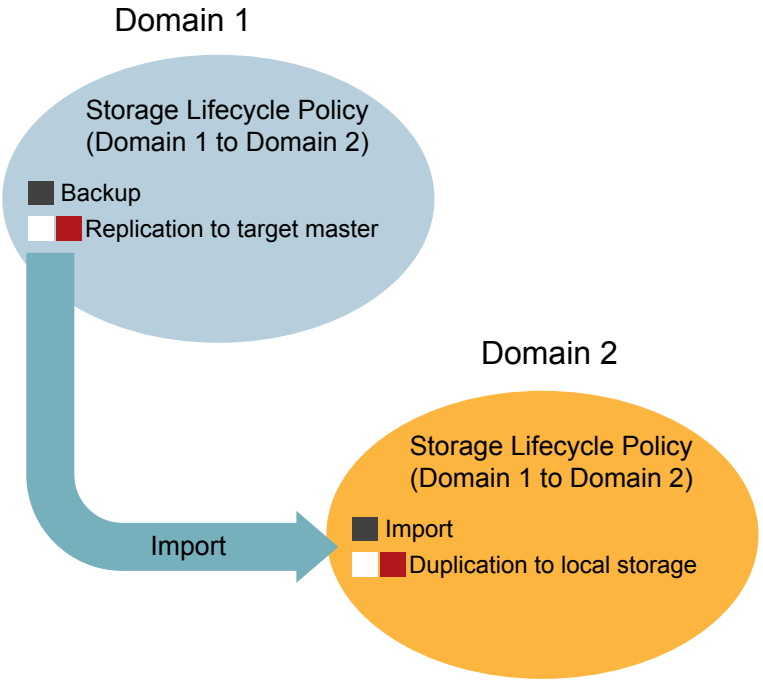
Event	Domain in which event occurs	Event description
1	The originating master server (Domain 1)	Clients are backed up according to a backup policy that indicates a storage lifecycle policy as the policy storage selection. After the backup, images are replicated from original domain to the target domain.

Table 2-2 Process overview of A.I.R. *(continued)*

Event	Domain in which event occurs	Event description
2	The target master server (Domain 2)	The storage server in the target domain recognizes that a replication event has occurred. It notifies the NetBackup master server in the target domain.
3	The target master server (Domain 2)	NetBackup imports the image immediately, based on an SLP that contains an import operation.
4	The target master server (Domain 2)	After the image is imported into the target domain, NetBackup continues to manage the copies in that domain.

Figure 2-1 is a typical A.I.R. setup that shows an image that is replicated from one source domain to one target domain.

Figure 2-1 Typical A.I.R. setup



Configuring Auto Image Replication (A.I.R.)

NetBackup provides the ability to establish a trust relationship between replication domains. A trust relationship is optional for an MSDP as the target storage.

The following items describe how a trust relationship affects A.I.R.:

- **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 only replicates to the specified domains rather than to all configured replication targets. This type of A.I.R. is known as targeted A.I.R.

To set up a master server for A.I.R.

1. On the master server of the source domain, open the **NetBackup Administration Console**, select **NetBackup Management** > **Host Properties** > **Master Servers**.
2. Double-click on the master server. In the **Master Server Properties** dialog box, in the left pane, click on **Servers**.
3. Select the **Trusted Master Servers** tab.
4. Click **Add**.
5. Add the name of the master server for the target domain in the **Trusted Master Server** field.
6. Click **Validate Certificate Authority**.
7. Use one of the following methods for authentication:
   - Select **Specify authentication token of the trusted master server** and then enter the token in the **Token** field.
     To create a token, review the *Creating authorization tokens* section in the NetBackup Security and Encryption Guide
   - Select **Specify credentials of the trusted master server** and then enter a **User name** and **Password** in the fields.
8. Click **OK** to complete the setup.

9 Repeat these steps in the target domain. Use the source master server name as the master server name in the **Validate Certificate Authority** field.

10 Configure storage server at both source domain and target domain.

The image is replicated from one storage server in the source domain to one storage server in the target domain. The image is needed to configure the MSDP at the source domain and the target domain. Use the Java GUI to configure the MSDP storage server, disk pool, and storage unit.

Deploying the certificate at storage server of source domain

MSDP supports secure communications between two media servers from two different NetBackup domains. The secure communication is set up when you run A.I.R.. The two media servers must use the same CA to do the certificate security check. The source MSDP server uses the Certificate Authority (CA) of the target NetBackup domain and the certificate that the target NetBackup domain authorized. You must manually deploy CA and the certificate on the source MSDP server before using A.I.R..

To configure the NetBackup CA and a NetBackup host ID-based certificate

1 On the source MSDP storage server, run the following command to get the NetBackup CA from target NetBackup master server:

- Windows:

```
install_path\NetBackup\bin
\nbcertcmd -getCACertificate -server target_master_server
```

- UNIX:

```
/usr/openv/netbackup/bin
/nbcertcmd -getCACertificate -server target_master_server
```

2 On the source MSDP storage server, run the following command to get the certificate generated by target NetBackup master server:

- Windows:

```
install_path\NetBackup\bin
\nbcertcmd -getCertificate
-server target_master_server -token token_string
```

- UNIX:

```
/usr/openv/netbackup/bin
/nbcertcmd -getCertificate
-server target_master_server -token token_string
```

Setting up the MSDP replication target

Images are replicated from source domain MSDP storage server to target domain MSDP storage server. The target MSDP server is the replication target of the source MSDP server. Use the Java GUI at the source domain to setup the replication target.

To set up the replication target

1. On the master server of the source domain, open the **NetBackup Administration Console**, select **Media and Device Management** > **Credentials** > **Storage Servers**.
2. Double-click on the source domain MSDP server.
3. In the **Replicaiton** tab, click on **Add**. Fill in the required information.

   The **Target storage server name** is the host name of the MSDP storage server in the target domain. The **User name** and **Password** is the credential used to configure the MSDP server in target domain.

Configuring a Storage Lifecycle Policy (SLP) for A.I.R.

To run a target A.I.R., you need to create an SLP at both the source domain and the target domain. Use Java GUI to create an import SLP.

Follow the procedures in Table 2-3 to configure the SLP.

Table 2-3 To configure an SLP

At target domain:	1	Open the **NetBackup Administration Console**, select **NetBackup Management** > **Storage** > **Storage Lifecycle Policies**.
	2	Click the **New Storage Lifecycle Policy** option, or right-click the blank area of the SLP list view and select the **New Storage Lifecycle Policy** to create an SLP.
	3	Type in the SLP name at the **New Storage Lifecycle Policy** dialog and then click **Add**.
	4	Select the **Import** option from the **Operation** drop down list.
	5	In the **Destination storage**, select the storage unit of the target MSDP storage server from the drop-down. Click **OK** option to complete the SLP creation.

Table 2-3 To configure an SLP *(continued)*

At source domain:	**1** Open the **NetBackup Administration Console**, select **NetBackup Management** > **Storage** > **Storage Lifecycle Policies**. **2** Click the **New Storage Lifecycle Policy** option, or right-click the blank area of the SLP list view and select the **New Storage Lifecycle Policy** to create an SLP. **3** Type in the SLP name at the **New Storage Lifecycle Policy** dialog and then click **Add**. **4** In the **New Operation** dialog, select the **Backup** option from the **Operation** drop down list. **5** In the **Destination storage**, select the storage unit of the source MSDP storage server from the drop-down. Click **OK**. **6** Click the newly added operation **Backup** item to highlight it and then click the **Add** option. **7** At the **New Operation** dialog select **Replication** option from the **Operation** drop down list. **8** Click the **A specific Master server** option item. Select the target master server from the **Target master server** drop down list. **9** Select the SLP name from the **Target import SLP** drop down list. Click **OK** on the **New Operation** dialog. **10** Click **OK** on the **New Storage Lifecycle Policy** dialog box.
Create a backup policy to perform a backup and run the SLP.	At the source domain, create a backup and use the SLP as Policy storage. Run the backup and after the backup runs, the replication job at the source domain runs. After a short period of time, the import job at the target domain runs. The target domain manages the replicated image at the target storage server.

Chapter 3

Planning your deployment

This chapter includes the following topics:

- Planning your MSDP deployment
- NetBackup naming conventions
- About MSDP deduplication nodes
- About the NetBackup deduplication destinations
- About MSDP storage capacity
- 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 performance
- About MSDP stream handlers
- MSDP deployment best practices

Planning your MSDP deployment

Table 3-1 provides an overview of planning your deployment of NetBackup deduplication.

Table 3-1 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 32. See "About the NetBackup deduplication destinations" on page 32.
Step 2	Understand the storage capacity and requirements	See "About MSDP storage capacity" on page 33. See "About MSDP storage and connectivity requirements" on page 34.
Step 3	Determine which type of deduplication to use	See "About NetBackup media server deduplication" on page 37. See "About NetBackup Client Direct deduplication" on page 42. See "About MSDP remote office client deduplication" on page 44.
Step 4	Determine the requirements for deduplication hosts	See "About MSDP storage servers" on page 39. See "About MSDP server requirements" on page 40. See "About MSDP client deduplication requirements and limitations" on page 44. See "About the network interface for MSDP" on page 46. See "About MSDP port usage" on page 47. See "About scaling MSDP" on page 55. See "About MSDP performance" on page 49.
Step 5	Determine the credentials for deduplication	See "About the NetBackup Deduplication Engine credentials" on page 45.
Step 6	Read about compression and encryption	See "About MSDP compression" on page 127. See "About MSDP encryption" on page 129.
Step 7	Read about optimized synthetic backups	See "About MSDP optimized synthetic backups" on page 48.
Step 8	Read about deduplication and SAN Client	See "About MSDP and SAN Client" on page 48.
Step 9	Read about optimized duplication and replication	See "About MSDP optimized duplication and replication" on page 49.
Step 10	Read about stream handlers	See "About MSDP stream handlers" on page 50.

Table 3-1 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 54.
Step 12	Determine the storage requirements and provision the storage	See "About provisioning the storage for MSDP" on page 60. See "About MSDP storage and connectivity requirements" on page 34. See "About MSDP storage capacity" on page 33. See "MSDP storage path properties" on page 110.
Step 13	License MSDP	See "About the MSDP license" on page 64. See "Licensing NetBackup MSDP" on page 65.
Step 14	Configure MSDP	See "Configuring MSDP server-side deduplication" on page 69. See "Configuring MSDP client-side deduplication" on page 71.
Step 15	Migrate from other storage to NetBackup deduplication	See "Migrating from another storage type to MSDP" on page 595.

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 (.)
- 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 45.

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 39.
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 40.
Storage	See "About the NetBackup deduplication destinations" on page 32.
Clients	The clients may include the clients that deduplicate their own data (Client Direct). See "About NetBackup Client Direct deduplication" on page 42.

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 37.

See "About MSDP storage servers" on page 39.

About the NetBackup deduplication destinations

Several destinations exist for the NetBackup deduplication, as shown in the following table.

Table 3-2 NetBackup deduplication storage destinations

Destination	Description
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.

About MSDP storage capacity

The MSDP storage contains one local LSU or multiple cloud LSUs. The following table describes the maximum deduplication storage capacity for a single **Media Server Deduplication Pool** that contains only one local LSU:

Table 3-3 Maximum MSDP storage capacities

Maximum capacity	Description
64 TBs	For all supported systems, NetBackup supports up to 64 TBs of storage in a single **Media Server Deduplication Pool**. See "About provisioning the storage for MSDP" on page 60.
400 TBs	NetBackup supports 400 TBs of storage in a new **Media Server Deduplication Pool** on the supported versions of the following operating systems: ■ Red Hat Linux ■ Windows Server ■ SUSE Linux Recommended operating systems: ■ Red Hat Linux 7.5 ■ Windows Server 2012 R2 Datacenter See "About provisioning the storage for MSDP" on page 60.

Table 3-3 Maximum MSDP storage capacities *(continued)*

Maximum capacity	Description
960 TBs	NetBackup 53xx appliances support up to 960TBs of storage in a single **Media Server Deduplication Pool**. See *About storage configuration* topic of the *NetBackup Appliance Administrator's Guide*.

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 32.

For the operating system versions that NetBackup supports for deduplication, see the NetBackup operating system compatibility list available through the following URL:

http://www.netbackup.com/compatibility

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. Veritas 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. Veritas 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.

NetBackup requires the exclusive use of the disk resources. If the storage is also used for purposes other than backups, NetBackup cannot manage disk pool capacity or manage storage lifecycle policies correctly. Therefore, NetBackup must be the only entity that uses the storage.

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:

http://www.netbackup.com/compatibility

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

See “About provisioning the storage for MSDP” on page 60.

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. Veritas supports iSCSI on storage networks with at least 10-Gigabit Ethernet network bandwidth. Veritas 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 36.

See "About MSDP storage capacity" on page 33.

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 3-4 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.

Table 3-4 Fibre Channel and iSCSI characteristics *(continued)*

Item	Fibre Channel	iSCSI
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, Veritas 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. Veritas 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 Veritas supports iSCSI for connectivity to **Media Server Deduplication Pool** storage, Veritas recommends Fibre Channel. Veritas 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 582.

See “MSDP media write error (84)” on page 584.

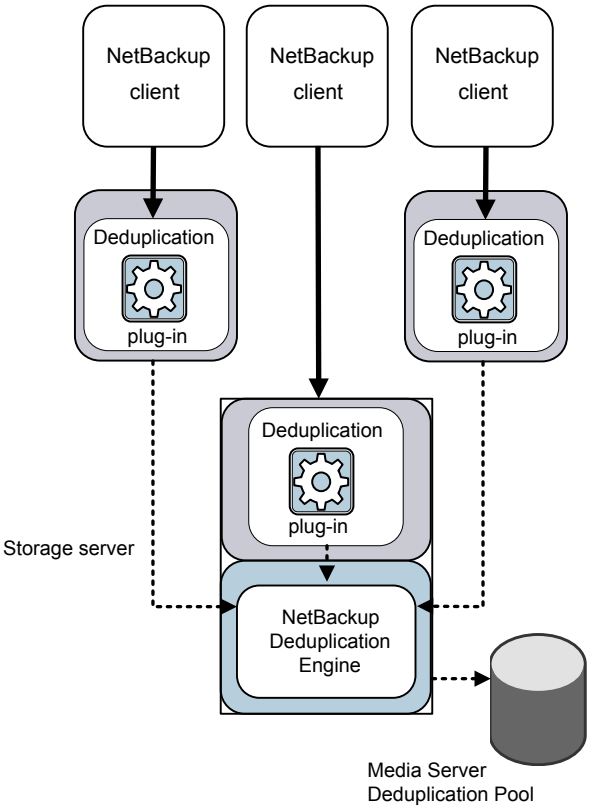
About NetBackup media server deduplication

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 3-1 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 3-1 NetBackup media server deduplication



More detailed information is available.

See “About MSDP deduplication nodes” on page 32.

See “About MSDP storage servers” on page 39.

See “About MSDP load balancing servers” on page 40.

See “About MSDP server requirements” on page 40.

See “About MSDP unsupported configurations” on page 42.

See “MSDP server components” on page 475.

See “Media server deduplication backup process” on page 478.

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 32.

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 42.
  See “About MSDP load balancing servers” on page 40.
- 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 141.

- 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 158.

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 56.

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.

See "About MSDP storage servers" on page 39.

See "About NetBackup media server deduplication" on page 37.

See "About MSDP storage servers" on page 39.

See "Managing MSDP servers" on page 422.

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 3-5 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 3-5 MSDP server minimum requirements

Component	Storage server	Load-balancing server
CPU	Veritas recommends at least a 2.2-GHz clock rate. A 64-bit processor is required. At least four cores are required. Veritas recommends eight cores. For 64 TBs of storage, Intel x86-64 architecture requires eight cores.	Veritas 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, Veritas recommends 1GB RAM for 1TB of storage. However if you go beyond 32 TBs of storage, Veritas 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 operating system compatibility list for your NetBackup release on the Veritas Support website. http://www.netbackup.com/compatibility	The operating system must be a supported 64-bit operating system. See the operating system compatibility list for your NetBackup release on the following website. http://www.netbackup.com/compatibility

A Veritas tech note provides detailed information about and examples for sizing the hosts for deduplication. Information includes the number of the NICs or the HBAs for each server to support your performance objectives.

For more information, refer to http://veritas.com/docs/TECH77575.

Note: This page has been updated for NetBackup version 7.5.

Note: In some environments, a single host can function as both a NetBackup master 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 master server operations.

See "About MSDP performance" on page 49.

See "About MSDP queue processing" on page 446.

About MSDP unsupported configurations

The following items describe some configurations that are not supported:

- NetBackup media server deduplication and Veritas 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.)

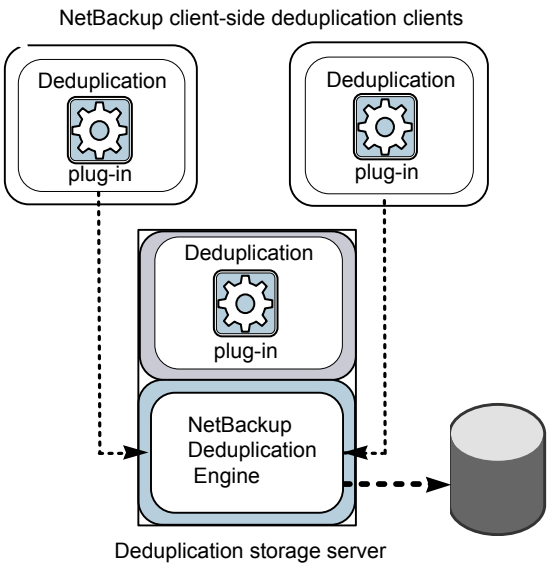
NetBackup Client Deduplication is a solution for the following cases:

- 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 44.
- 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 3-2 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 3-2 NetBackup client deduplication



More information is available.

See “About MSDP client deduplication requirements and limitations” on page 44.

See “About MSDP remote office client deduplication” on page 44.

See “MSDP client components” on page 479.

See “MSDP client–side deduplication backup process” on page 480.

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 compatibility lists at the following URL:

http://www.netbackup.com/compatibility

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, Veritas recommends that the clients and the servers be at the same release and revision.

The *NetBackup Release Notes* is available through the following URL:

http://www.veritas.com/docs/DOC5332

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 199.

See “Specifying resilient connections” on page 203.

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

See “MSDP pd.conf file parameters” on page 209.

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 83.

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 129.

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 458.

About remote client backup scheduling

NetBackup backup policies use the time zone of the master server for scheduling jobs. If your remote clients are in a different time zone than your NetBackup master server, you must compensate for the difference. For example, suppose the master 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 (@)

Note: Record and save the credentials in case you need them in the future.

Caution: 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 Veritas 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 113. See "Configuring a storage server for a Media Server Deduplication Pool" on page 95.
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 139. See "Configuring a separate network path for MSDP duplication and replication" on page 140.

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

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 3-6 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.

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 139.

In NetBackup, the **OptimizedImage** attribute enables optimized synthetic backups. It applies to both storage servers and deduplication pools. Beginning with NetBackup 7.1, the **OptimizedImage** attribute is enabled by default on storage servers and media server deduplication pools. For the storage servers and the disk pools that you created in NetBackup releases earlier than 7.1, you must set the **OptimizedImage** attribute on them so they support optimized synthetic backups.

See "Setting MSDP storage server attributes" on page 424.

See "Setting a Media Server Deduplication Pool attribute" on page 436.

Table 3-7 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 3-8 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 141. See "About MSDP cloud support" on page 256.
To a remote NetBackup domain	See "About NetBackup Auto Image Replication" on page 158.

About MSDP performance

Many factors affect performance, especially the server hardware and the network capacity.

Table 3-9 provides information about performance during backup jobs for a deduplication storage server. The deduplication storage server conforms to the minimum host requirements. Client deduplication or load balancing servers are not used.

See “About MSDP server requirements” on page 40.

Table 3-9 MSDP job load performance for an MSDP storage server

When	Description
Normal operation	Normal operation is when all clients have been backed up once. Approximately 15 to 20 jobs can run concurrently and with high performance under the following conditions: ■ The hardware meets minimum requirements. (More capable hardware improves performance.) ■ No compression. If data is compressed, the CPU usage increases quickly, which reduces the number of concurrent jobs that can be handled. ■ The deduplication rate is between 50% and 100%. The deduplication rate is the percentage of data already stored so it is not stored again. ■ The amount of data that is stored is between 30% to 90% of the capacity of the storage.
Storage approaches full capacity	NetBackup maintains the same number of concurrent backup jobs as during normal operation under the following conditions: ■ The hardware meets minimum requirements. (More capable hardware improves performance.) ■ The amount of data that is stored is between 85% to 90% of the capacity of the storage. However, the average time to complete the jobs increases significantly.

How file size may affect the MSDP deduplication rate

The small file sizes that are combined with large file segment sizes may result in low initial deduplication rates. However, after the deduplication engine performs file fingerprint processing, deduplication rates improve. For example, a second backup of a client shortly after the first does not show high deduplication rates. But the deduplication rate improves if the second backup occurs after the file fingerprint processing.

How long it takes the NetBackup Deduplication Engine to process the file fingerprints varies.

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.

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

Oracle stream handler

The Oracle stream handler is not enabled by default for existing and new Oracle clients in NetBackup 8.3. Also, the Oracle stream handler only supports stream-based backups and you can enable and disable the Oracle stream handler per `<client> <policy>` combination using the `cacontrol` command line utility.

In NetBackup 10.0, 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*`\Veritas\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.

In NetBackup 8.3, 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 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>
```

Note: When you use `cacontrol` to set *<POLICY>* or *<STHTYPE>* to `enabled`, NetBackup enables all the old clients which have existing images. The deduplication rate decreases significantly only at the first backup after enabled. Also, the storage usage increases only in the first backup after enabled. Basically, NetBackup behaves as if you have run a first full backup. Both the deduplication rate and storage usage improve after initial activation of the stream handler.

When using the `cacontrol` command utility to create a `Marker Entry` in NetBackup 10.0, 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, 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

You can apply the Microsoft SQL Server stream handler to all of the Microsoft SQL Server version and Azure SQL Server. You can use the **MS-SQL** policy or the **Standard** policy to enable this feature.

You can enable and disable the Microsoft SQL Server stream handler per policy or all policies at once using the `cacontrol` command line utility.

The marker entry configuration file (`marker.cfg`) is used to override the default behavior by using the`cacontrol` command utility with the `--sth` flag at a client and or policy level only.

The `marker.cfg` file is stored at the following location:

`/MDSP_SERVER/databases/spa/marker.cfg`

Update the `marker.cfg` file by using the following `cacontrol` options:

- You can create and or update the `marker.cfg` file.

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

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

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

- You can delete the settings to use the default behavior.

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

When you enable the Microsoft SQL Server stream handler, the **Job Details** tab in the Administration Console displays the following:

MS-SQL stream handler enabled

MSDP deployment best practices

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

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

Use fully qualified domain names

Veritas 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 584.

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.

Veritas 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 80.

Veritas 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

Veritas 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.

See "About MSDP performance" on page 49.

Introduce MSDP load balancing servers gradually

Veritas 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.

Many factors affect deduplication server performance.

See "About MSDP performance" on page 49.

Because of the various factors, Veritas 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.

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 127.

See "About MSDP encryption" on page 129.

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.

More detailed information about backup streams is available.

http://www.veritas.com/docs/TECH77575

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. Veritas recommends that you select the **Failover** method for the **Storage unit selection** type.

Table 3-10 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 141. ■ 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 48.

About protecting the MSDP data

Veritas 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 148.

- 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 156.

Veritas also recommends that you back up the MSDP catalog.

See "About protecting the MSDP catalog" on page 231.

Save the MSDP storage server configuration

Veritas 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 225.

See "Saving the MSDP storage server configuration" on page 226.

See "Editing an MSDP storage server configuration file" on page 227.

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, Veritas recommends that you disable the hardware caches. Read and write performance may decline, but you help to avoid data loss.

Chapter 4

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 32.

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

See “About MSDP optimized duplication and replication” on page 49.

Other NetBackup requirements may affect how you provision the storage.

See “About MSDP storage and connectivity requirements” on page 34.

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

See "About the NetBackup deduplication destinations" on page 32.

See "Planning your MSDP deployment" on page 30.

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 storage capacity" on page 33.

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.
  Veritas recommends that you use a descriptive naming convention for the mount point names.

Steps to configure the 400 TB MSDP using 32 TB volumes

1. Create, format, and mount 9 new file systems - one file system must have 1 TB storage space and the other 8 file systems must have 32 TB storage space each.
2. Mount the 1 TB file system at `/msdp/cat` and the 32 TB file systems on `/msdp/vol0`, `/msdp/vol1` and so on until each volume is mounted.
3. Create a touch a 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 MSDP through the **Storage Server Configuration Wizard**. 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 32 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
```

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 456.

Do not modify MSDP storage directories and files

Unless you are directed to do so by the NetBackup documentation or by a Veritas 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, Veritas recommends that you use the Veritas InfoScale Storage. InfoScale Storage includes the Veritas Volume Manager and the Veritas File System.

For supported systems, see the InfoScale hardware compatibility list at the Veritas 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 5

Licensing deduplication

This chapter includes the following topics:

- About the MSDP license
- Licensing NetBackup MSDP

About the MSDP license

NetBackup deduplication is licensed separately from base NetBackup.

The license enables both NetBackup media server deduplication and NetBackup client deduplication. The license is a front-end capacity license. It is based on the size of the data to be backed up, not on the size of the deduplicated data.

If you remove the license or if it expires, you cannot create new deduplication disk pools. you also cannot create the storage units that reference NetBackup deduplication pools. NetBackup does not delete the disk pools or the storage units that reference the disk pools. You can use them again if you enter a valid license.

The license also enables the **Use Accelerator** feature on the NetBackup policy **Attributes** tab. Accelerator increases the speed of full backups for files systems. Accelerator works with deduplication storage units as well as with other storage units that do not require the deduplication option. More information about Accelerator is available.

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

http://www.veritas.com/docs/DOC5332

Before you try to install or upgrade to a NetBackup version that supports deduplication, you should determine on which operating systems Veritas supports deduplication. See the NetBackup operating system compatibility list:

http://www.netbackup.com/compatibility

See “Licensing NetBackup MSDP” on page 65.

Licensing NetBackup MSDP

If you installed the license for deduplication when you installed or upgraded NetBackup, you do not need to perform this procedure.

Enter the license on the NetBackup master server. The following procedure describes how to use the **NetBackup Administration Console** to enter the license key.

To license NetBackup MSDP

1. On the **Help** menu of the **NetBackup Administration Console** on the NetBackup master server, select **License Keys**.
2. In the **NetBackup License Keys** dialog box, click **New**.
3. In the **Add a New License Key** dialog box, enter the license key and click **Add** or **OK**.
4. In the **NetBackup License Key** dialog box, click **Close**.
5. Restart all the NetBackup services and daemons.

Chapter 6

Configuring deduplication

This chapter includes the following topics:

- Configuring MSDP server-side deduplication
- Configuring MSDP client-side deduplication
- About the MSDP Deduplication Multi-Threaded Agent
- Configuring the Deduplication Multi-Threaded Agent behavior
- Configuring deduplication plug-in interaction with the Multi-Threaded Agent
- About MSDP fingerprinting
- About the MSDP fingerprint cache
- Configuring the MSDP fingerprint cache behavior
- About seeding the MSDP fingerprint cache for remote client deduplication
- Configuring MSDP fingerprint cache seeding on the client
- Configuring MSDP fingerprint cache seeding on the storage server
- About sampling and predictive cache
- Enabling 400 TB support for MSDP
- About MSDP Encryption using NetBackup KMS service
- About MSDP Encryption using external KMS server
- Configuring a storage server for a Media Server Deduplication Pool
- About disk pools for NetBackup deduplication
- Configuring a disk pool for deduplication

- Creating the data directories for 400 TB MSDP support
- Adding volumes to a 400 TB Media Server Deduplication Pool
- Configuring a Media Server Deduplication Pool storage unit
- Configuring client attributes for MSDP client-side deduplication
- Disabling MSDP client-side deduplication for a client
- About MSDP compression
- About MSDP encryption
- MSDP compression and encryption settings matrix
- Configuring encryption for MSDP backups
- Configuring encryption for MSDP optimized duplication and replication
- About the rolling data conversion mechanism for MSDP
- Modes of rolling data conversion
- MSDP encryption behavior and compatibilities
- Configuring optimized synthetic backups for MSDP
- About a separate network path for MSDP duplication and replication
- Configuring a separate network path for MSDP duplication and replication
- About MSDP optimized duplication within the same domain
- Configuring MSDP optimized duplication within the same NetBackup 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 MSDP cloud
- About storage lifecycle policies
- About the storage lifecycle policies required for Auto Image Replication
- Creating a storage lifecycle policy
- About MSDP backup policy configuration

- Creating a backup policy
- Resilient Network properties
- Specifying resilient connections
- Adding an MSDP load balancing server
- About variable-length deduplication on NetBackup clients
- About the MSDP pd.conf configuration file
- Editing the MSDP pd.conf file
- About the MSDP contentrouter.cfg file
- About saving the MSDP storage server configuration
- Saving the MSDP storage server configuration
- Editing an MSDP storage server configuration file
- 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
- Changing the MSDP shadow catalog path
- Changing the MSDP shadow catalog schedule
- Changing the number of MSDP catalog shadow copies
- Configuring an MSDP catalog backup
- Updating an MSDP catalog backup policy
- 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

Configuring MSDP server-side deduplication

This topic describes how to configure media server deduplication in NetBackup.

Table 6-1 describes the configuration tasks.

The *NetBackup Administrator's Guide* describes how to configure a base NetBackup environment.

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

http://www.veritas.com/docs/DOC5332

Table 6-1 MSDP configuration tasks

Step	Task	Procedure
Step 1	Install the license for deduplication	See "Licensing NetBackup MSDP" on page 65.
Step 2	Create NetBackup log file directories on the master server and the media servers	See "NetBackup MSDP log files" on page 567. See " Creating NetBackup log file directories for MSDP " on page 566.
Step 3	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 72. See "Configuring the Deduplication Multi-Threaded Agent behavior" on page 74. See "Configuring deduplication plug-in interaction with the Multi-Threaded Agent" on page 79.
Step 4	Configure the fingerprint cache behavior	Configuring the fingerprint cache behavior is optional. See "About the MSDP fingerprint cache" on page 81. See "Configuring the MSDP fingerprint cache behavior" on page 82.
Step 5	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 90.

Table 6-1 MSDP configuration tasks *(continued)*

Step	Task	Procedure
Step 6	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 wizard also lets you configure a disk pool and a storage unit. See "About MSDP storage servers" on page 39. See "MSDP storage path properties" on page 110. See "About MSDP optimized duplication and replication" on page 49. Which type of storage server to configure depends on the storage destination. See "About the NetBackup deduplication destinations" on page 32. See "Configuring a storage server for a Media Server Deduplication Pool" on page 95.
Step 7	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 113. See "Configuring a disk pool for deduplication" on page 114.
Step 8	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 118.
Step 9	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 118.
Step 10	Configure a storage unit	See "Configuring a Media Server Deduplication Pool storage unit" on page 121.
Step 11	Enable encryption	Encryption is optional. See "Configuring encryption for MSDP backups" on page 132.
Step 12	Configure optimized synthetic backups	Optimized synthetic backups are optional. See "Configuring optimized synthetic backups for MSDP" on page 139.

Table 6-1 MSDP configuration tasks *(continued)*

Step	Task	Procedure
Step 13	Configure MSDP restore behavior	Optionally, you can configure NetBackup to bypass media servers during restores. See "How MSDP restores work" on page 457. See " Configuring MSDP restores directly to a client " on page 458.
Step 14	Configure optimized duplication copy	Optimized duplication is optional. See "About MSDP optimized duplication within the same domain" on page 141.
Step 15	Configure replication	Replication is optional. See "About MSDP replication to a different domain" on page 155.
Step 16	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 198. See "Creating a backup policy" on page 199.
Step 17	Specify advanced deduplication settings	Advanced settings are optional. See "About the MSDP pd.conf configuration file" on page 208. See "Editing the MSDP pd.conf file" on page 209. See "MSDP pd.conf file parameters" on page 209.
Step 18	Protect the MSDP data and catalog	See "About protecting the MSDP data" on page 58. See "About protecting the MSDP catalog" on page 231.

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 MSDP server-side deduplication" on page 69.

Table 6-2 Client deduplication configuration tasks

Step	Task	Procedure
Step 1	Configure media server deduplication	See "Configuring MSDP server-side deduplication" on page 69.

Table 6-2 Client deduplication configuration tasks *(continued)*

Step	Task	Procedure
Step 2	Learn about client deduplication	See "About NetBackup Client Direct deduplication" on page 42.
Step 3	Configure a resilient connection for remote office clients	Resilient connections are optional. See "About MSDP remote office client deduplication" on page 44. See "Resilient Network properties" on page 199. See "Specifying resilient connections" on page 203.
Step 4	Enable client-side deduplication	See "Configuring client attributes for MSDP client-side deduplication" on page 125.
Step 5	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 85. See "About seeding the MSDP fingerprint cache for remote client deduplication" on page 83. See "Configuring MSDP fingerprint cache seeding on the storage server" on page 86.
Step 6	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 458.

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 6-3 Interactions and behaviors

Interaction	Procedure
Multi-Threaded Agent behavior and resource usage	See "Configuring the Deduplication Multi-Threaded Agent behavior" on page 74.
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 79.
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 79.
The backup policies that should use the Deduplication Multi-Threaded Agent	See "Configuring deduplication plug-in interaction with the Multi-Threaded Agent" on page 79.

Table 6-4 describes the operational notes for MSDP multithreading. If the Multi-Threaded Agent is not used, NetBackup uses the single-threaded mode.

Table 6-4 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 ■ NetBackup Accelerator ■ `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 209.

Table 6-4 Multi-Threaded Agent requirements and limitations *(continued)*

Item	Description
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. Veritas recommends that you use the MSDP compression and encryption rather than NetBackup policy-based compression and encryption. See "About MSDP compression" on page 127. See "About MSDP encryption" on page 129.

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 72.

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 75.

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 74.

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 6-5 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`

Table 6-5 Logging parameters (mtstrm.conf file) *(continued)*

Logging Parameter	Description

`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 6-6 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 209. 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 6-7). (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 209. Possible values: 1-32, inclusive. **Warning:** Veritas 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).
`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.

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

Process Parameter	Description
`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 6-7 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 209.
`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 208.

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 209.

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.

NetBackup 8.0 and previous versions use fingerprinting calculations that are based on the MD5-like algorithm. Starting with NetBackup 8.1, the fingerprinting calculations are based on a more secure SHA-2 algorithm. On a system that is upgraded to the 8.1 version, every new segment is computed with the SHA-2 algorithm. A data rolling conversion task works in the background to convert the existing MD5-like fingerprints to SHA-2 fingerprints, gradually.

See "About the rolling data conversion mechanism for MSDP" on page 135.

NetBackup 8 .1 can handle both fingerprint types, and the new server is compatible with old clients and old servers. When you perform a backup from an old client to a new server or when you duplicate data from an old server to a new server, conversion from MD5-like to SHA-2 occurs inline on the new server before the data is saved to the disk. Similarly, when you duplicate data from a new server to an old server, conversion from SHA-2 to MD5-like occurs inline on the new server before the data is sent to the old server.

Notes and restrictions that there are some known issues for the compatibility support.

- The fingerprint conversion requires additional computation time. The interaction between old clients and old servers and new server is slower than if both the client and the server are new.
- You cannot restore data that is backed up using SHA-2 algorithm on a media server that uses the MD5-like algorithm. However, you may choose to restore the SHA-2 fingerprint data on a new media server.
- Similarly, you cannot use client-direct restore to restore data that is backed up using Client Direct deduplication on a media server that uses the MD5-like algorithm. However, you may choose to restore the data on a new media server.
- If you are using two types of media servers for load balancing, where one media server uses MD5-like algorithm and the other media server uses the SHA-2 algorithm, the initial backup may lose deduplication ratio. Therefore, split the old media servers and the new media servers into different groups, and create different storage unit for each of them.
- When data is backed up from a NetBackup 7.5 or previous version client, using Client Direct deduplication, most of the data is transferred over the network and deduplicated on the server. This may save storage, but it does not reduce network throughput. It is recommended that you upgrade the NetBackup client to the latest version.

See "About the MSDP fingerprint cache" on page 81.

See "Media server deduplication backup process" on page 478.

See "MSDP client–side deduplication backup process" on page 480.

About the MSDP fingerprint cache

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

See "About MSDP fingerprinting" on page 80.

The storage server maintains an index cache of the fingerprints in RAM. For each backup job, a client requests a list of the fingerprints from its last backup from the server.

The NetBackup Deduplication Engine (`spoold`) loads a percentage of the fingerprints into the cache at startup. After startup, the Engine loads the remaining fingerprints.

You can configure the cache loading behavior.

See "Configuring the MSDP fingerprint cache behavior" on page 82.

You can also control the fingerprint cache seeding for clients.

See “About seeding the MSDP fingerprint cache for remote client deduplication” on page 83.

Configuring the MSDP fingerprint cache behavior

You can configure the cache loading behavior.

See “About the MSDP fingerprint cache” on page 81.

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 82.

MSDP fingerprint cache behavior options

Table 6-8 describes the parameters that control the behavior. All of these options are in the `contentrouter.cfg` file.

The parameters are stored in the `contentrouter.cfg` file.

See “About the MSDP contentrouter.cfg file” on page 224.

Table 6-8 Cache load parameters

Behavior	Description
`CacheLoadThreadNum`	The number of threads to use to load the remaining fingerprints. The `CacheLoadThreadNum` in the `contentrouter.cfg` file controls the number of threads. NetBackup begins loading fingerprints from the next container number after the startup fingerprint loading. The default is one.
`MaxCacheSize`	The percentage of RAM to use for the fingerprint cache. The `MaxCacheSize` in the `contentrouter.cfg` file controls percentage of RAM. The default is 50%.

About seeding the MSDP fingerprint cache for remote client deduplication

Veritas 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 6-9 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 85.
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 86.

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 567.

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 567.

For more information about seeding, see the following Veritas tech note:

http://www.veritas.com/docs/TECH144437

See "About the MSDP fingerprint cache" on page 81.

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 83.

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 209.

  See "MSDP pd.conf file parameters" on page 209.

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 83.

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 88.

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, Veritas 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 is 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 cluster deployments including Flex Scale, AKS, and EKS deployment. This method is also enabled for cloud-only volumes for MSDP non-cluster platforms that are NetBackup appliance, Flex, and BYO. 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 Cache section of configuration file `contentrouter.cfg`.

The default values for non-cluster deployments:

Configuration	Default value
`MaxCacheSize`	50%
`MaxPredictiveCacheSize`	20% (10% in NetBackup Appliance)
`MaxSamplingCacheSize`	5% (10% in NetBackup Appliance)
`EnableLocalPredictiveSamplingCache` in `contentrouter.cfg`	`false`
`EnableLocalPredictiveSamplingCache` in `spa.cfg`	`false`

The default values for cluster deployments:

Configuration	Default value
`MaxCacheSize`	512MiB
`MaxPredictiveCacheSize`	40%
`MaxSamplingCacheSize`	20%
`EnableLocalPredictiveSamplingCache` in `spa.cfg`	`true`
`EnableLocalPredictiveSamplingCache` in `contentrouter.cfg`	`true`

For MSDP cluster deployments, the local volume and cloud volume share the same S-cache and P-cache size. For the non-cluster deployment, S-cache and P-cache are only for cloud volume, and `MaxCacheSize` is still used for local volume. In case the system is not used for cloud backup, `MaxPredictiveCacheSize` and `MaxSamplingCacheSize` can be set to a small value, for example, 1% or 128MiB. `MaxCacheSize` can be set to a large value, for example, 50% or 60%. Similarly, if the system is used for cloud backups only, `MaxCacheSize` can be set to 1% or 128MiB, and `MaxPredictiveCacheSize` and `MaxSamplingCacheSize` can be set to a larger value.

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 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 stream needs 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.

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 storage capacity” on page 33.

See “About provisioning the storage for MSDP” on page 60.

For additional configuration information, refer to the following article:

How to configure a 400 TB Media Server Deduplication Pool (MSDP) on Linux and Windows

About MSDP Encryption using NetBackup KMS service

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

MSDP encryption carries out segment-level encryption and assigns a unique encryption key for every data segment. A customer key is retrieved from NetBackup KMS to encrypt the segment key.

Key creation and activation actions must be done manually (or using scripts) by the user.

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

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

If the KMS service is not available for MSDP or the key in the KMS service that is used by MSDP is not available, then MSDP waits in an infinite loop. When MSDP goes in an infinite loop, few 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.

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 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 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 enabling KMS, refer to the following topics:

See "Configuring a storage server for a Media Server Deduplication Pool" on page 95.

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 7.7.3 to 8.1.2 or later
- 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 *Veritas 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 master 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 master server hostname.

After completing the steps, you can upgrade MSDP.

Enabled KMS encryption for Local LSU

To enable KMS encryption configurations for local LSU, you can create a configuration file and then run the `nbdevconfig` command.

Configuration file contents for updating encryption configurations are as follows:

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 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.
`V7.5 "keygroupname" "" string`	Key group name must have valid characters: A-Z, a-z, 0-9, _ (underscore), - (hyphen), : (colon), . (period), and space.

Example to enable KMS status for local LSU:

```
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
```

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 KMS server.

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 encrypt the backup data.

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 KMS service" on page 91.

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 39.

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 45.
The storage paths.	See "MSDP storage path properties" on page 110.
The network interface.	See "About the network interface for MSDP" on page 46.
The load-balancing servers, if any.	See "About MSDP storage servers" on page 39.

When you configure the storage server, the wizard 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 118.

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

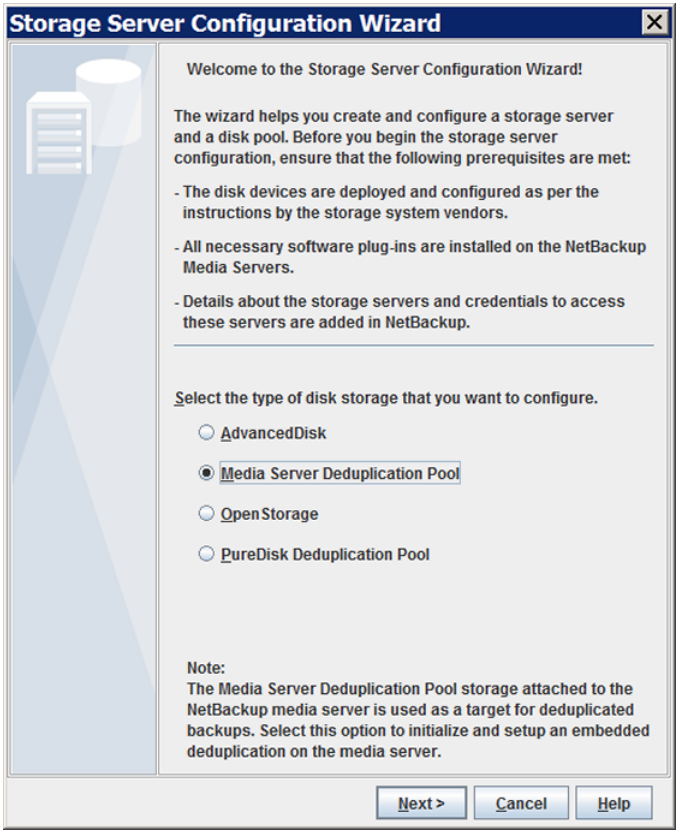
1. In the **NetBackup Administration Console**, select either **NetBackup Management** or **Media and Device Management**.
2. In the right pane, click **Configure Disk Storage Servers**.

   The **Storage Server Configuration Wizard Welcome** panel appears.

3 On the **Welcome** panel, select **Media Server Deduplication Pool** from the drop-down menu.

The types of storage servers that you can configure depend on the options for which you are licensed.

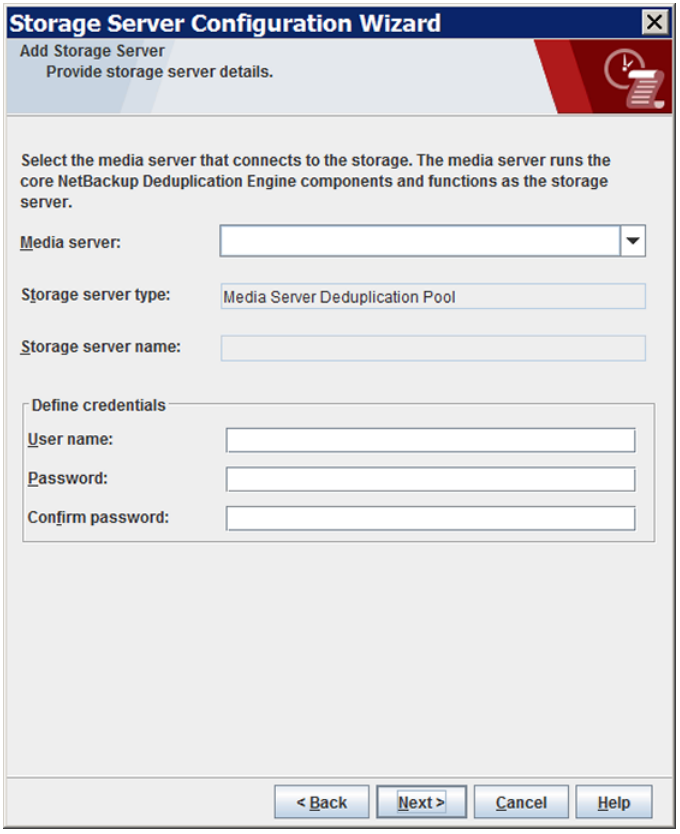
The following is an example of the wizard panel:



After you select **Media Server Deduplication Pool**, click **Next**. The **Add Storage Server** wizard panel appears.

4 On the **Add Storage Server** panel, select or enter the appropriate information.

The following is an example of the wizard panel:



Media server Select the media server that you want to configure as the storage server.

You can add deduplication load-balancing servers on the next wizard panel.

Username Enter the user name for the NetBackup Deduplication Engine.

See "About the NetBackup Deduplication Engine credentials" on page 45.

Password Enter the password for the NetBackup Deduplication Engine.

Confirm password To confirm the password, re-enter the password.

After you enter the information, click **Next**.

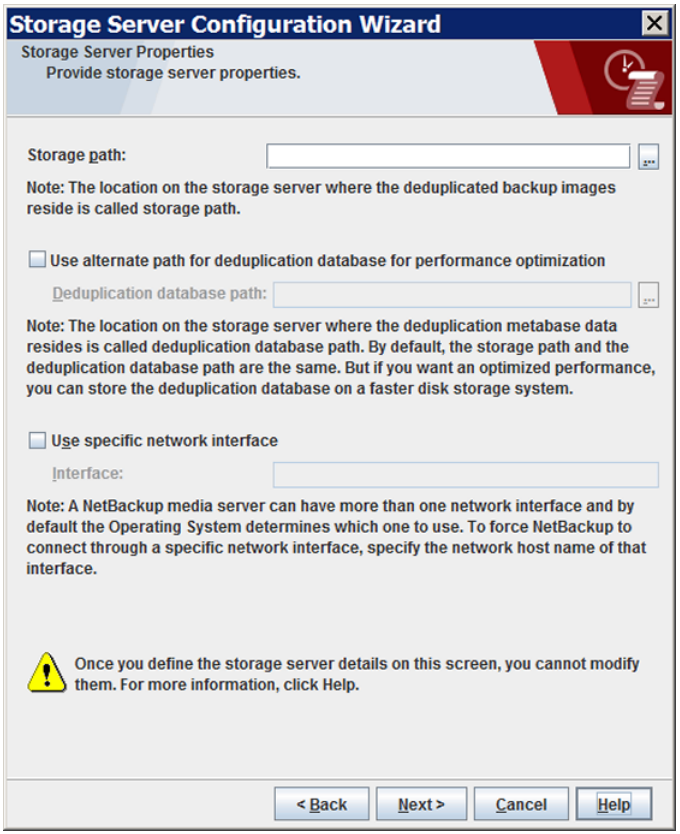
The **Storage Server Properties** panel appears.

5 On the **Storage Server Properties** panel, enter or select the properties for the deduplication storage server.

See "MSDP storage path properties" on page 110.

See "MSDP network interface properties" on page 113.

The following is an example of the wizard panel:



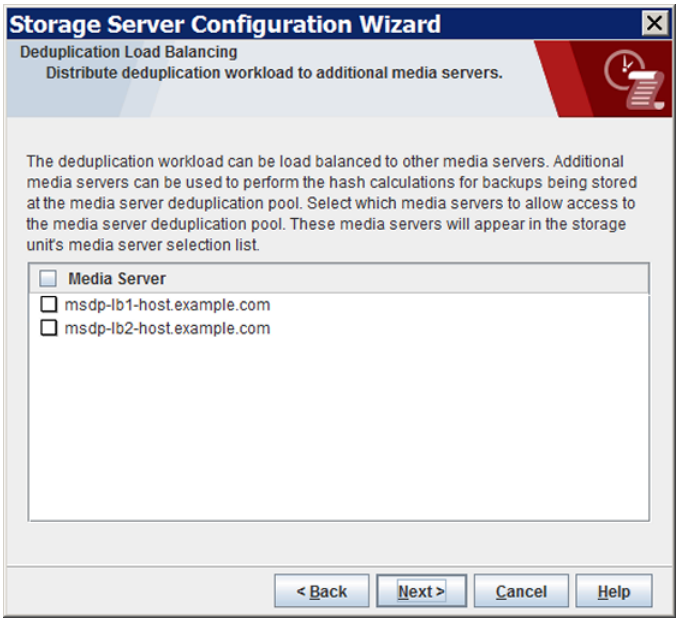
After you click **Next**, the behavior depends on whether you have media servers installed, as follows:

No media servers installed	The **Storage Server Encryption** panel appears. Go to step 8.
Media servers are installed.	The **Deduplication Load Balancing** panel appears. Continue to step 6.

6 On the **Deduplication Load Balancing** panel, select other NetBackup media servers to use for deduplication. Selecting load-balancing servers is optional.

See "About MSDP load balancing servers" on page 40.

The following is an example of the wizard panel:



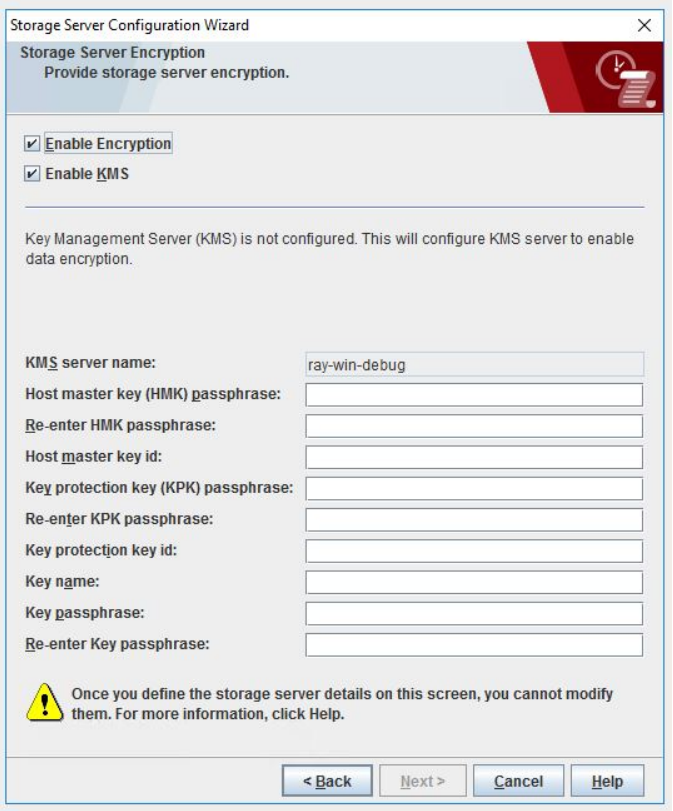
For the media servers that you select, NetBackup enables their deduplication functionality and populates them with the NetBackup Deduplication Engine credentials you entered on a previous panel.

When you click **Next**, the **Storage Server Encryption** panel appears next.

7 On the **Storage Server Encryption** panel, you can enable encryption and KMS for the Media Server Deduplication Pool. The **Enable KMS** option is available when you select **Enable Encryption**.

If servers are configured, NetBackup KMS is configured:

The following is an example of the wizard panel:



See "About MSDP encryption" on page 129.

See "About MSDP Encryption using NetBackup KMS service" on page 91.

See "About MSDP Encryption using external KMS server" on page 95.

- If you select **Enable KMS** for the first time, as a one-time KMS configuration, you must enter the following information:

Option	Description
KMS server name	The name of the Key Management Server.
Host master key (HMK) passphrase	Enter a password that you want to set as the host master key (HMK) passphrase.

Option	Description
Host master key id	Enter a unique ID to associate with the host master key. This ID helps you to determine an HMK associated with any keystore.
Key protection key (KPK) passphrase	Enter a password that you want to set as the key protection key (KPK) passphrase. This ID helps you to determine a KPK associated with any keystore.
Key protection key id	Enter a unique ID to associate with the key protection key.
Key name	Enter a name of the key.
Key passphrase	Enter a password that you want set as the key passphrase.

- If you select **Enable KMS** and if NetBackup KMS is already configured on the Master Server, you must enter the following information:

Option	Description
Key name	Enter a name of the key.
Key passphrase	Enter a password that you want set as the key passphrase.

- If you select **Enable KMS** and if external KMS is already configured on the Master Server, you must enter the following information:

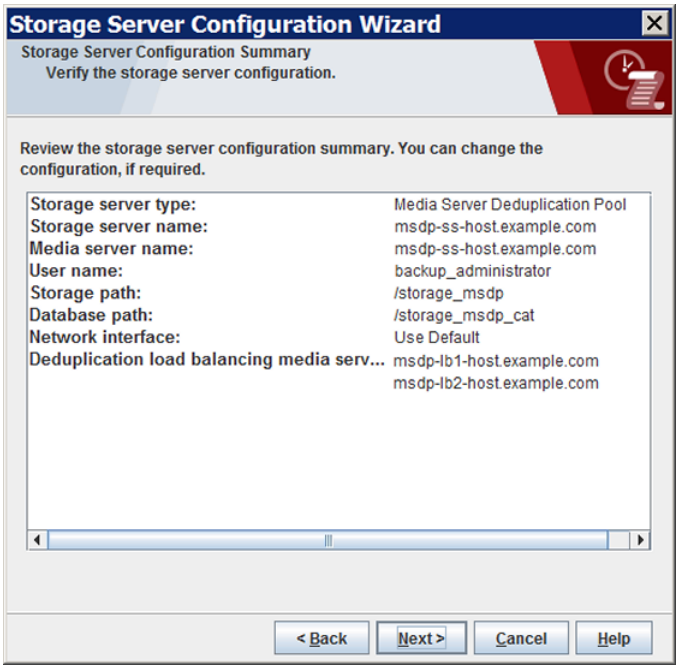
Option	Description
Key group name	Enter a name of the key group.

  Ensure that a key with a custom attribute set with value of key group name already exists in the external KMS server.
  For more information about KMS configuration, see the NetBackup Security and Encryption Guide.

When you click **Next**, the **Storage Server Configuration Summary** panel appears next.

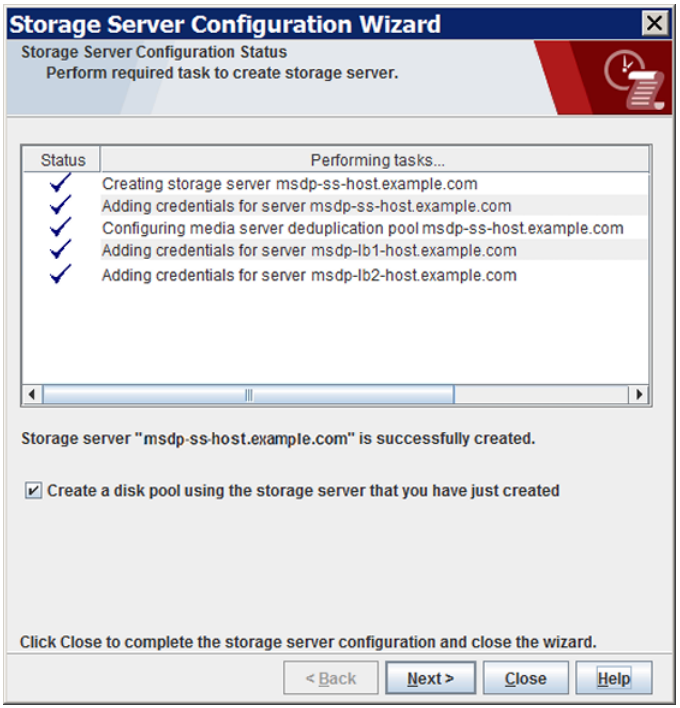
8 On the **Storage Server Configuration Summary** panel, verify the selections. If OK, click **Next** to configure the storage server.

The following is an example of the wizard panel:



The **Storage Server Creation Status** panel appears.

9 The **Storage Server Configuration Status** wizard panel describes the status of the operation.



After the storage server is created, you can do the following:

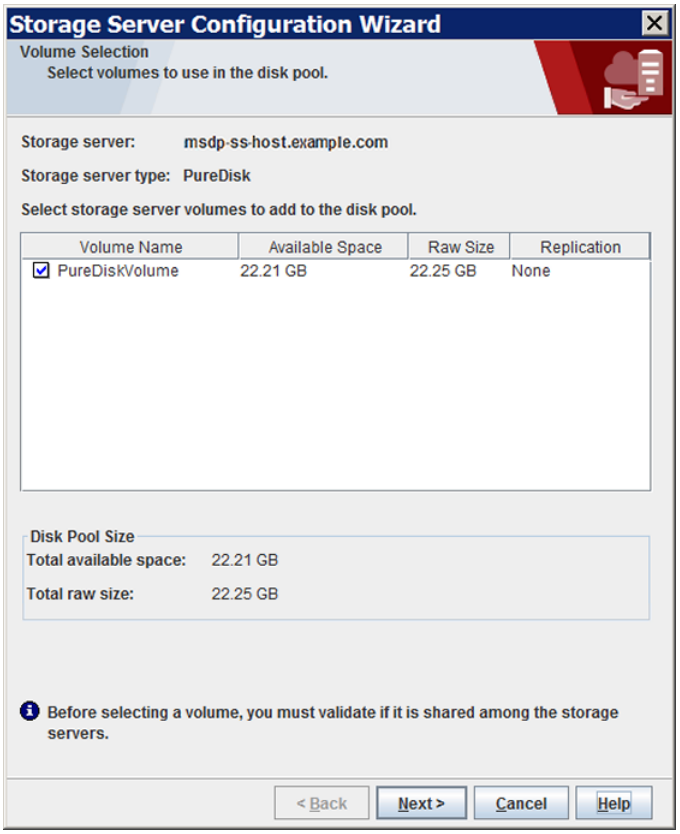
Configure a disk pool	Ensure that **Create a disk pool using the storage server that you have just created** is selected and then click **Next**. The **Select Volumes** panel appears. Continue to the next step.
Exit	Click **Close**. You can configure a disk pool at another time.

If storage server creation fails, see the following:

See “Troubleshooting MSDP configuration issues” on page 574.

10 On the **Select Volumes** panel, select the volume for this disk pool. All of storage in the **Storage Path** that you configured in the **Storage Server Configuration Wizard** is exposed as a single volume. The **PureDiskVolume** is a virtual name for that storage.

The following is an example of the wizard panel:

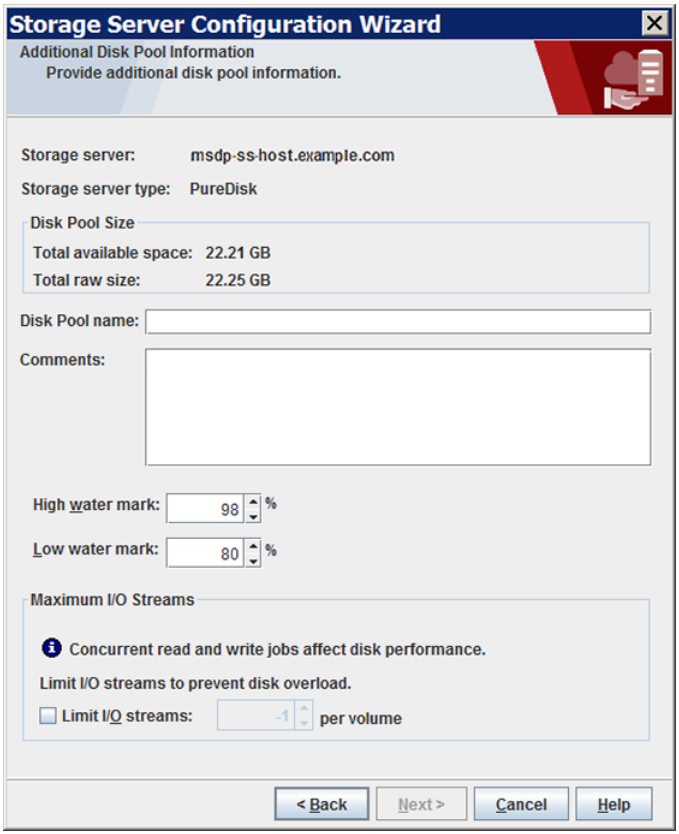


After you select the **PureDiskVolume** volume, click **Next**. The **Additional Disk Pool Information** wizard panel appears.

11 On the **Additional Disk Pool Information** panel, enter the values for this disk pool.

See "Media Server Deduplication Pool properties" on page 116.

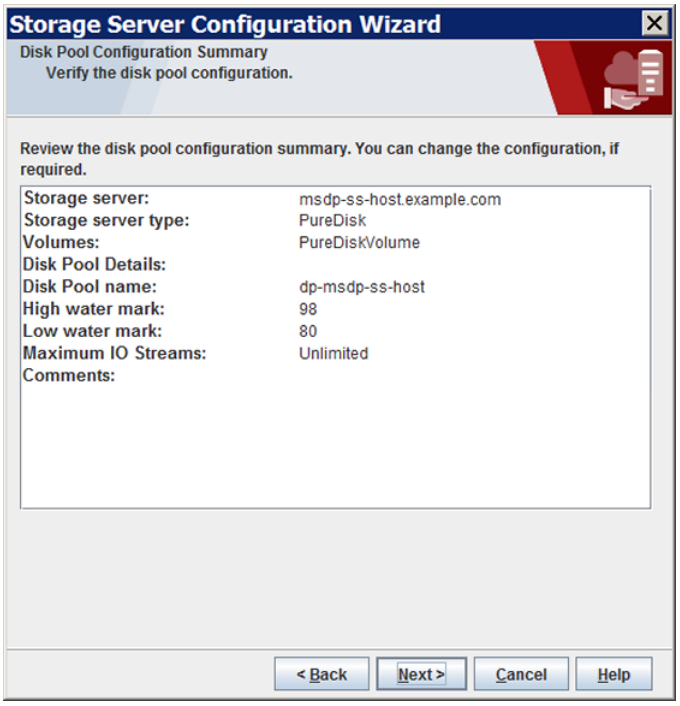
The following is an example of the wizard panel:



After you enter the appropriate information or select the necessary options, click **Next**. The **Disk Pool Configuration Summary** wizard panel appears.

12 On the **Disk Pool Configuration Summary** panel, verify the selections. If OK, click **Next**.

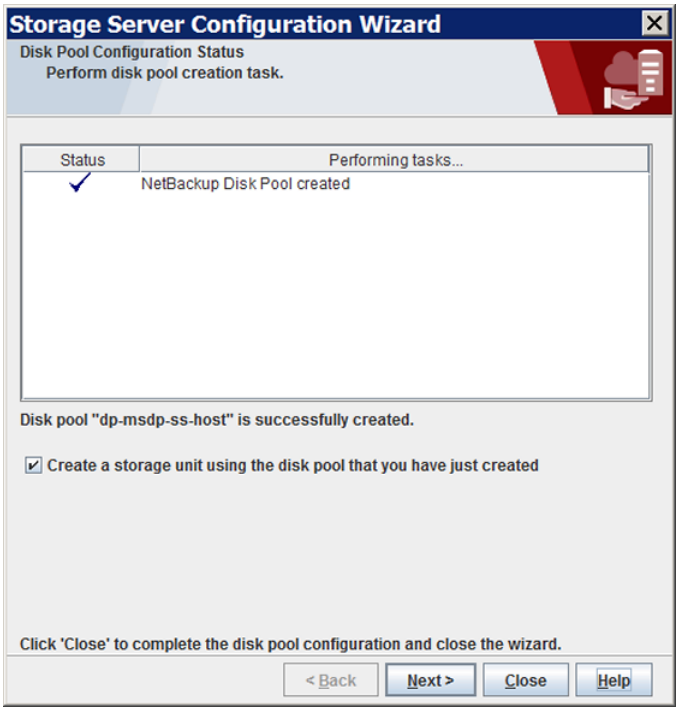
The following is an example of the wizard panel:



To configure the disk pool, click **Next**. The **Disk Pool Configuration Status** wizard panel appears.

13 The **Disk Pool Configuration Status** wizard panel describes the progress of the operation.

The following is an example of the wizard panel:



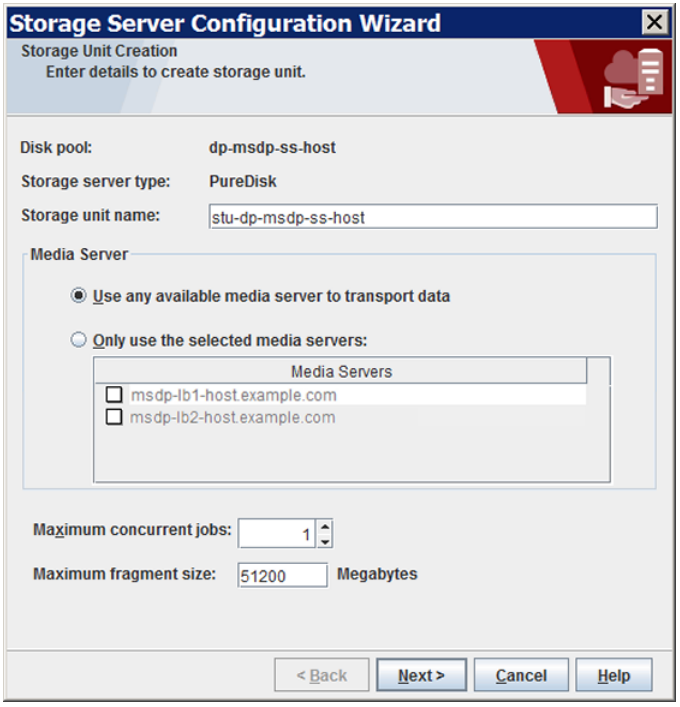
After the disk pool is created, you can do the following:

Configure a storage unit	Ensure that **Create a storage unit using the disk pool that you have just created** is selected and then click **Next**. The **Storage Unit Creation** wizard panel appears. Continue to the next step.
Exit	Click **Close**. You can configure one or more storage units later. See "Configuring a Media Server Deduplication Pool storage unit" on page 121.

14 Enter the appropriate information for the storage unit.

See "Media Server Deduplication Pool storage unit properties" on page 122.

The following is an example of the wizard panel:



After you enter the appropriate information or select the necessary options, click **Next** to create the storage unit.

15 After NetBackup configures the storage unit, the **Finished** panel appears. Click **Finish** to exit from the wizard.

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 6-10 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 "About MSDP storage capacity" on page 33.
	
	See "About provisioning the storage for MSDP" on page 60.
	
	See "Creating the data directories for 400 TB MSDP support" on page 118.
	
	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 "About MSDP storage and connectivity requirements" on page 34.

Table 6-10 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 60. 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 "About provisioning the storage for MSDP" on page 60. 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.

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 6-11 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 46.
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 6-12 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 141.
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 162. 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 158.

Configuring a disk pool for deduplication

The NetBackup **Storage Server Configuration Wizard** lets you configure one disk pool during storage server configuration. To configure additional disk pools, launch the **Disk Pool Configuration Wizard**. Before you can configure a NetBackup disk pool, a NetBackup deduplication storage server must exist.

See "About disk pools for NetBackup deduplication" on page 113.

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

- The type of disk pool:
  - A **Media Server Deduplication Pool** on the disk storage that is attached to a NetBackup deduplication media server.
- 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.
  See "Media Server Deduplication Pool properties" on page 116.

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

To configure a deduplication disk pool by using the wizard

1. In the **NetBackup Administration Console**, select either **NetBackup Management** or **Media and Device Management**.
2. From the list of wizards in the right pane, click **Configure Disk Pool**.
3. Click **Next** on the welcome panel of the wizard.

   The **Disk Pool Configuration Wizard** panel appears.
4. On the **Disk Pool Configuration Wizard** panel, select the type of disk pool you want to configure in the **Storage server type** window.

   The types of disk pools that you can configure depend on the options for which you are licensed.

   After you select the disk pool in the **Storage server type** window, click **Next**.
5. On the **Storage Server Selection** panel, select the storage server for this disk pool. The wizard displays the deduplication storage servers that are configured in your environment.

   Click **Next**.
6. On the **Volume Selection** panel, select the volume for this disk pool.

Media Server Deduplication Pool	All of storage in the **Storage Path** that you configured in the **Storage Server Configuration Wizard** is exposed as a single volume. The **PureDiskVolume** is a virtual name for that storage.

   After you select the volume, click **Next**.
7. On the **Additional Disk Pool Information** panel, enter the values for this disk pool.

   See "Media Server Deduplication Pool properties" on page 116.

   After you enter the appropriate information or select the necessary options, click **Next**.
8. On the **Disk Pool Configuration Summary** panel, verify the selections. If OK, click **Next**.

   To configure the disk pool, click **Next**.

9 The **Disk Pool Configuration Status** panel describes the progress of the operation.

After the disk pool is created, you can do the following:

Configure a storage unit	Ensure that **Create a storage unit using the disk pool that you have just created** is selected and then click **Next**. The **Storage Unit Creation** wizard panel appears. Continue to the next step.
Exit	Click **Close**. You can configure one or more storage units later. See "Configuring a Media Server Deduplication Pool storage unit" on page 121.

10 In the **Storage Unit Creation** panel, enter the appropriate information for the storage unit.

See "Media Server Deduplication Pool storage unit properties" on page 122.

After you enter the appropriate information or select the necessary options, click **Next** to create the storage unit.

11 After NetBackup configures the storage unit, the **Finished** panel appears. Click **Finish** to exit from the wizard.

Media Server Deduplication Pool properties

Table 6-13 describes the disk pool properties.

Table 6-13 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.

Table 6-13 Media server deduplication pool properties *(continued)*

Property	Description
Total raw size	The total raw size of the storage in the disk pool.
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.

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 Configuration Wizard**.)

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 60.
- The storage server must be configured already.
  See "Configuring a storage server for a Media Server Deduplication Pool" on page 95.

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. (Example: VMware Backup Host, NDMP Backup Agent, media server is also a primary server.
- **Swap:** 64 GB
- **Storage:**
  - **Metadata disk:** RAID 0+1 is recommended, with at least 1 TB of space.
  - Veritas 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 60.

See "Configuring a storage server for a Media Server Deduplication Pool" on page 95.

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 Configure the MSDP through the **Storage Server Configuration Wizard** and **Ensure that the Use alternate path for deduplication database** option is checked.

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
```

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:). Veritas 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 storage capacity" on page 33.

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 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 113.

The **Disk Pool Configuration Wizard** lets you create a storage unit; therefore, 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 **NetBackup Management > Storage > Storage Units** window of the Administration Console.

To configure a storage unit from the Actions menu

1. In the **NetBackup Administration Console**, expand **NetBackup Management > Storage > Storage Units**.
2. On the **Actions** menu, select **New > Storage Unit**.
3. Complete the fields in the **New Storage Unit** dialog box.

   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 122.

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 6-14 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**.
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.

Table 6-14 Media Server Deduplication Pool storage unit properties *(continued)*

Property	Description
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, Veritas recommends that you use the default, maximum fragment size to ensure optimal deduplication performance. For more information, see the *NetBackup Snapshot Client Administrator's Guide*: http://www.veritas.com/docs/DOC5332

Table 6-14 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.

MSDP storage unit recommendations

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

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.

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.

Configuring client attributes for MSDP client-side deduplication

To configure client deduplication, set an attribute in the NetBackup master 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. In the **NetBackup Administration Console**, expand **NetBackup Management > Host Properties > Master Servers**.
2. In the details pane, select the master server.
3. On the **Actions** menu, select **Properties**.
4. Select the **Client Attributes** properties.
5. On the **General** tab of the **Client Attributes** properties, add the clients that you want to deduplicate their own data to the **Clients** list, as follows:
   - Click **Add**.
   - In the **Add Client** dialog box, enter a client name or browse to select a client. Then click **Add**.
     Repeat for each client that you want to add.
   - When you finish adding clients, click **Close**.
6. 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 **Disable client-side deduplication** in the *NetBackup Administrator's Guide, Volume I*:

   http://www.veritas.com/docs/DOC5332

See "Disabling MSDP client-side deduplication for a client" on page 127.

See "About NetBackup Client Direct deduplication" on page 42.

See "About the NetBackup deduplication options" on page 16.

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. In the NetBackup Administration Console, expand **NetBackup Management > Host Properties > Master Servers**.
2. In the details pane, select the master server.
3. On the **Actions** menu, select **Properties**.
4. On the **Host Properties Client Attributes General** tab, select the client that deduplicates its own data.
5. In the **Deduplication Location** drop-down list, select **Always use the media server**.
6. Click **OK**.

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 6-15 describes the compression options.

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

See “MSDP compression and encryption settings matrix” on page 130.

Table 6-15 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 72.
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 208.

See "Use MSDP compression and encryption" on page 57.

About MSDP encryption

NetBackup provides encryption for the deduplicated data. It is separate from and different than NetBackup policy-based encryption. By default, MSDP encryption is disabled.

Table 6-16 describes the encryption options.

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

See "MSDP compression and encryption settings matrix" on page 130.

Table 6-16 MSDP encryption options

Option	Description
Backup encryption	For backups, the deduplication plug-in encrypts the data after it is deduplicated. The data remains encrypted 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 MSDP `pd.conf` file `ENCRYPTION` parameter controls backup encryption for individual hosts. By default, backup encryption is disabled on all MSDP hosts. If you want backup encryption, you must enable it on the following MSDP hosts: ■ The clients that deduplicate their own data (that is, client-side deduplication). ■ The MSDP load balancing servers. ■ The MSDP storage server. See "Configuring encryption for MSDP backups" on page 132. **Note:** Do not enable backup encryption by selecting the **Encryption** option on the **Attributes** tab of the **Policy** dialog box. If you do, NetBackup encrypts 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 72.

Table 6-16 MSDP encryption options *(continued)*

Option	Description
Duplication and replication encryption	For duplication and replication, the deduplication plug-in on MSDP servers encrypts the data for transfer. The data is encrypted during transfer from the plug-in to the NetBackup Deduplication Engine on the target storage server and remains encrypted on the target storage. The MSDP `pd.conf` file `OPTDUP_ENCRYPTION` parameter controls duplication and replication encryption for individual hosts. By default, duplication and replication encryption is disabled on the MSDP storage server and on the MSDP load balancing servers. If you want duplication and replication encryption, you must enable it on the following MSDP servers: ■ The load balancing servers. ■ The storage server. Duplication and replication encryption does not apply to clients. NetBackup chooses the least busy host to initiate and manage each duplication job and replication job. See "Configuring encryption for MSDP optimized duplication and replication" on page 134.

NetBackup 8.0 introduced 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 rolling data conversion mechanism for MSDP" on page 135.

See "MSDP encryption behavior and compatibilities" on page 138.

MSDP compression and encryption settings matrix

Four MSDP `pd.conf` file parameters control the compression and the encryption for individual hosts. Table 6-17 shows the matrix for the outcomes for the compression and the encryption parameters.

Table 6-17 Compression and encryption settings and outcomes

Parameters	Default: Compress both the backups and duplication and replication	Settings A: Compress and encrypt the backups	Settings B: Compress the backups and compress and encrypt duplication and replication	Settings C: Compress and encrypt backups and compress duplication and replication	Settings D: Compress and encrypt both backups and duplication and replication
ENCRYPTION	0	1	0	1	1
COMPRESSION	1	1	1	1	1
OPTDUP_ENCRYPTION	0	0	1	0	1
OPTDUP_COMPRESSION	1	0	1	1	1
Outcomes on the disk pools	Backup MSDP: Encryption: 0 Compression: 1 Target MSDP: Encryption: 0 Compression: 1	Backup MSDP: Encryption: 1 Compression: 1 Target MSDP: Encryption: 1 Compression: 1	Backup MSDP: Encryption: 0 Compression: 1 Target MSDP: Encryption: 1 Compression: 1	Backup MSDP: Encryption: 1 Compression: 1 Target MSDP: Encryption: 1 Compression: 1	Backup MSDP: Encryption: 1 Compression: 1 Target MSDP Encryption: 1 Compression: 1
Notes		See the following note.		See the following note.	See the following note.

Note: Settings A and C have the same outcome on target storage as setting D because of the interaction of compression and encryption for the backups. If the backups are encrypted and compressed, they are also encrypted and compressed during optimized duplication and replication and encrypted and compressed on target storage. The OPTDUP_ENCRYPTION setting is ignored.

For client-side deduplication clients, a parameter on the storage server can override the pd.conf file ENCRYPTION parameter.

See "Configuring encryption for MSDP backups" on page 132.

See "About MSDP compression" on page 127.

See "About MSDP encryption" on page 129.

See "About the MSDP pd.conf configuration file" on page 208.

Configuring encryption for MSDP backups

Two procedures exist to configure encryption during backups for MSDP, as follows:

Configure encryption on individual hosts running OST plug-in	Use this procedure to configure encryption on the MSDP hosts that are running OST plug-in. The `ENCRYPTION` parameter in the MSDP `pd.conf` file controls encryption for that host. The parameter applies only to the host on which you modify the `pd.conf`, as follows: See "To configure backup encryption on a single host" on page 133. To have all data in MSDP server encrypted, you must change pd.conf parameter of all load-balancing media servers and client hosts. If you want to encrypt all data in the MSDP pool, do not use this method. Use the MSDP server configuration instead. Using `pd.conf` to change MSDP encryption is deprecated. It is recommended that you use the server option instead.
Configure encryption for all backups to the MSDP server	Use this procedure to configure encryption to have all data to the MSDP server encrypted. If you use this procedure, you do not have to configure any hosts that send data to this MSDP server including NetBackup media server, servers in opt-dup, servers in AIR, and client direct hosts. The `ServerOptions` parameter in the MSDP `contentrouter.cfg` file controls encryption for all hosts that send data to this MSDP server. This parameter supersedes the `pd.conf` file `ENCRYPTION` setting. See "To configure backup encryption for all backups targeted to this MSDP pool" on page 132.

To configure backup encryption for all backups targeted to this MSDP pool

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. The following line is an 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.

To ensure that encryption occurs for all backups jobs, configure it on all MSDP hosts. MSDP hosts include the MSDP storage server, the MSDP load-balancing servers, and the NetBackup Client Direct deduplication clients.

See "About MSDP encryption" on page 129.

If you want to encrypt all data in the MSDP pool, do not use the following method due to its complexity. It is recommended that you use the server option instead.

To configure backup encryption on a single host

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

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 209.

2 For the line that begins with `#ENCRYPTION`, remove the pound sign (or hash sign, `#`) in column 1.

3 In that same line, replace the `0` (zero) with a `1`.

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 On the client-side deduplication clients and on the MSDP load-balancing servers, ensure that the `LOCAL_SETTINGS` parameter in the `pd.conf` file is set to `1`. Doing so ensures that the setting on the current host has precedence over the server setting.

5 Save and close the file.

6 If the host is the storage server or a load-balancing server, restart the NetBackup Remote Manager and Monitor Service (`nbrmms`) on the host.

Configuring encryption for MSDP optimized duplication and replication

The OPTDUP_ENCRYPTION parameter in the pd.conf file on the MSDP host controls duplication and replication encryption for that host. The data that is encrypted during transfer remains encrypted on the target storage.

If you want to encrypt all data in the MSDP pool, it is recommended that you use the server option. pd.conf file controls only the source host encryption behavior.

Use this procedure to configure encryption for optimized duplication and replication on MSDP storage servers and on MSDP load balancing servers. To ensure that encryption occurs for all optimized duplication and replication jobs, configure it on all MSDP servers.

By default, optimized duplication encryption is disabled on all MSDP hosts.

See "About MSDP encryption" on page 129.

To configure replication and duplication encryption on an MSDP server

1. Use a text editor to open the pd.conf file on the MSDP server.

   The pd.conf file resides in the following directories:

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

2. For the line that begins with #OPTDUP_ENCRYPTION, remove the pound sign (or hash sign, #) in column 1.

3. In that same line, replace the 0 (zero) with a 1.

   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.

   See "MSDP pd.conf file parameters" on page 209.

4. On load balancing servers, ensure that the LOCAL_SETTINGS parameter is set to 1. Doing so ensures that the ENCRYPTION setting on the load balancing server is used.

5. Save and close the file.

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

About the rolling data conversion mechanism for MSDP

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.

In NetBackup 8.1, with the introduction of the AES and the SHA-2 algorithms, we want to convert the data that is encrypted and computed with the older algorithms (Blowfish and MD5-like) to the newer algorithms (AES-256 and SHA-512/256).

The environments that are upgraded to NetBackup 8.1 may include Blowfish encrypted data and the MD5-like fingerprints that need to be converted to the new format. To handle the conversion and secure the data, a new internal task converts the current data container to the AES-256 encryption and the SHA-512/256 fingerprint algorithm. This new task is referred to as the rolling data conversion. The conversion begins automatically after an upgrade to NetBackup 8.0. You can control some aspects of the conversion process or stop it entirely.

Rolling data conversion traverses all existing data containers. If the data is encrypted with the Blowfish algorithm, the data is re-encrypted with the AES-256 algorithm. Then a new SHA-512/256 fingerprint is generated. After the conversion, the data container has an additional `.map` file, in addition to the `.bhd` and `.bin` files. The `.map` file contains the mapping between the SHA-512/256 and the MD5-like fingerprints. It is used for the compatibility between SHA-512/256 fingerprints and MD5-like fingerprints. The `.bhd` file includes the SHA-512/256 fingerprints.

When you upgrade to NetBackup 8.1.1, there might be encrypted data that is not encrypted using a customer key. The encrypted data must be encrypted by a customer key and to handle the data conversion and secure the data, a new internal task encrypts the existing data using a customer key. After the encryption and the fingerprint rolling conversion completes, the KMS rolling conversion begins.

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.

Veritas 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.

  Veritas 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 Veritas 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 6-18 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 `--dataconverton` 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`
`--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.

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 6-19 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
Load balancing server with NetBackup version earlier than 8.0	AES (using inline data conversion)

Table 6-20 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.

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 "Creating a backup policy" on page 199.

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

   http://www.veritas.com/docs/DOC5332

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

See "Setting MSDP storage server attributes" on page 424.

See "Creating a backup policy" on page 199.

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 46.

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 140. 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 141.

See "About MSDP replication to a different domain" on page 155.

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 139.

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
   ```

   Veritas 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
   ```

   Veritas 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 master 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 152.

You can use a separate network for the duplication traffic.

See "About a separate network path for MSDP duplication and replication" on page 139.

See "Configuring MSDP optimized duplication within the same NetBackup domain" on page 148.

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 143.
- 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*:
  http://www.veritas.com/docs/DOC5332
- 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 152.

- 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 Administration Console**.
- 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.

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.

About MSDP push duplication within the same domain

Figure 6-1 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 6-1 Push duplication environment

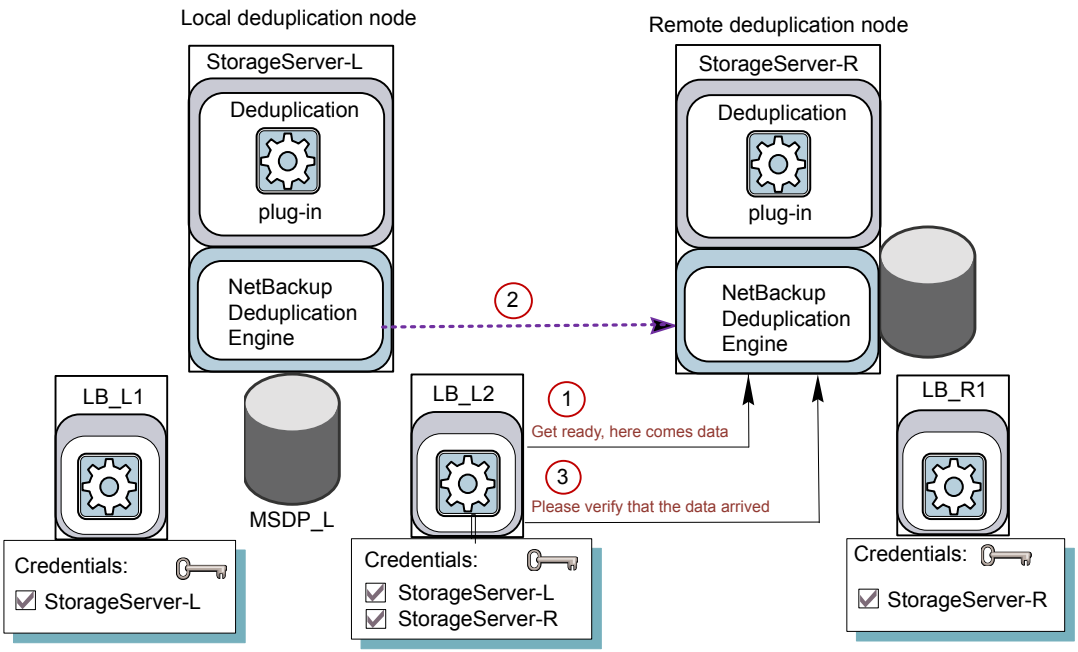


Figure 6-2 shows the settings for the storage unit for the normal backups for the local deduplication node. The disk pool is the **MSDP_L** in the local environment.

Because all hosts in the local node are co-located, you can use any available media server for the backups.

Figure 6-2 Storage unit settings for backups to MSDP_L

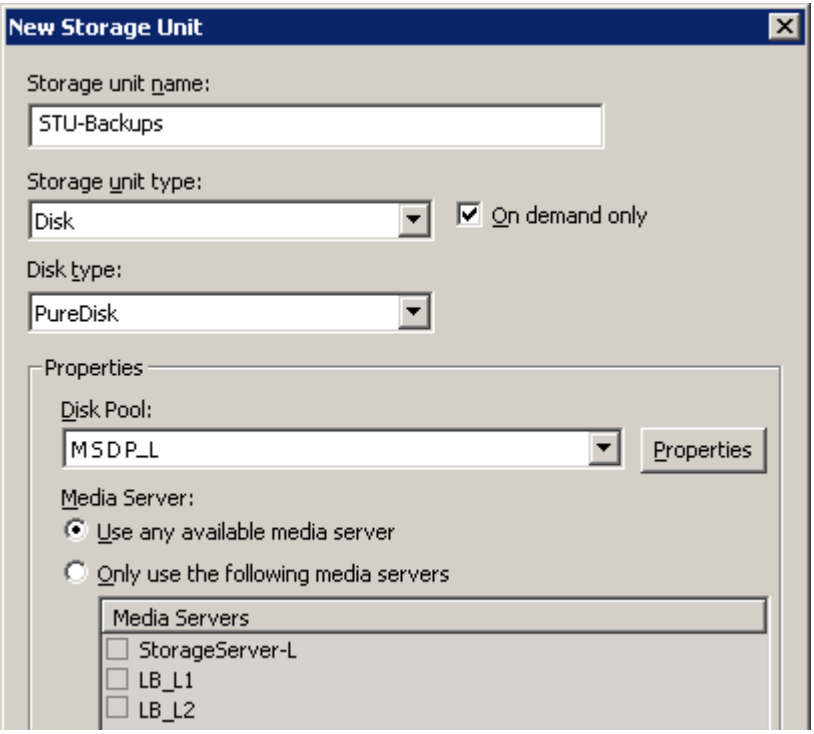
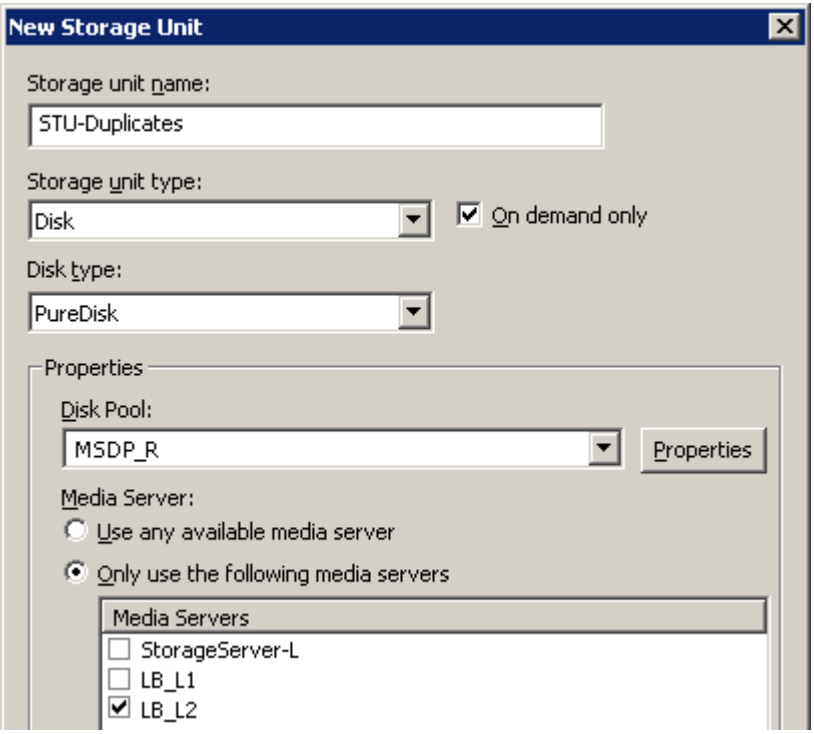


Figure 6-3 shows the storage unit settings for the optimized duplication. The destination is the **MSDP_R** in the remote environment. You must select the common server, so only load balancing server LB_L2 is selected.

Figure 6-3 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.

You can use a load balancing server when you duplicate between two NetBackup deduplication pools.

About MSDP pull duplication within the same domain

Figure 6-4 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 6-4 Pull duplication

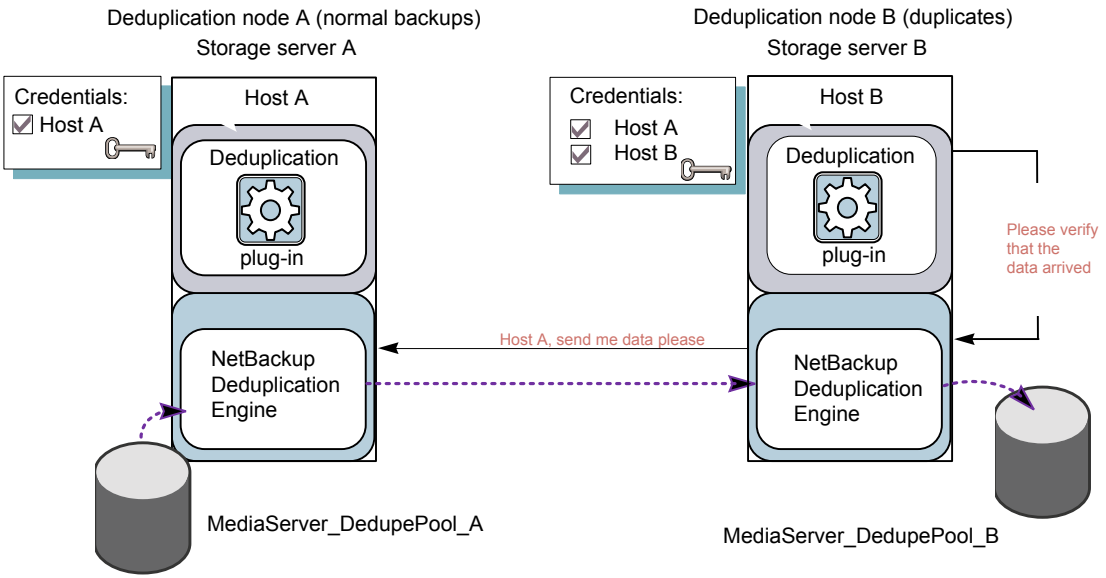
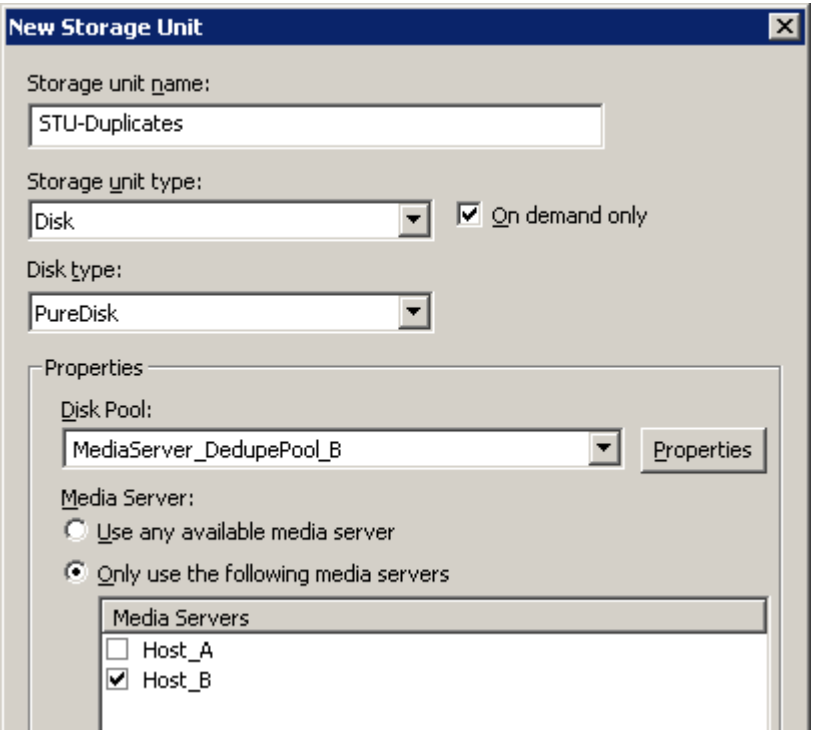


Figure 6-5 shows the storage unit settings for the duplication destination. They are similar to the push example except host B is selected. Host B is the common server, so it must be selected in the storage unit.

Figure 6-5 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 6-21 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 141.

Table 6-21 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 95. 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 143. 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 Configuration Wizard** to configure them. See "Configuring a disk pool for deduplication" on page 114.
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.) See "Configuring a Media Server Deduplication Pool storage unit" on page 121.

Table 6-21 How to configure optimized duplication of deduplicated data *(continued)*

Step	Action	Description
Step 5	Configure the storage unit for duplication	Veritas 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. See "Configuring a Media Server Deduplication Pool storage unit" on page 121.
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 188.
Step 7	Configure optimized duplication behaviors	Optionally, you can configure the optimized duplication behavior. See "Configuring NetBackup optimized duplication or replication behavior" on page 152. See "About configuring MSDP optimized duplication and replication bandwidth" on page 188.

Table 6-21 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 189. See "Creating a storage lifecycle policy" on page 192.
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 198. See "Creating a backup policy" on page 199.

Table 6-21 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*: http://www.veritas.com/docs/DOC5332

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 6-22 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 153. ■ 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 154.
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 154.

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 master 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 master 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 master 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 154.

  Alternatively on UNIX systems, you can add the entry to the bp.conf file on the NetBackup master server.

To configure the storage lifecycle policy wait period

1. in the **NetBackup Administration Console**, expand **NetBackup Management** > **Host Properties** > **Master Servers**. Select the master server, and then on the **Actions** menu click **Properties**.
2. Select **SLP Parameters**.
3. Change the **Extended image retry interval** to the new value.
4. Click **OK**.

Setting NetBackup configuration options by using the command line

Veritas recommends that you use the **NetBackup Administration Console Host Properties** to configure NetBackup properties.

However, some properties cannot be set by using the **Administration Console**. 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 6-23 describes the MSDP replication source and targets that NetBackup supports.

Table 6-23 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 152.

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 139.

See "Configuring MSDP replication to a different NetBackup domain" on page 156.

See "About MSDP optimized duplication and replication" on page 49.

Configuring MSDP replication to a different NetBackup domain

Table 6-24 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 139.

Table 6-24 NetBackup MSDP replication configuration tasks

Step	Task	Procedure
Step 1	Learn about MSDP replication	See "About MSDP replication to a different domain" on page 155. See "About NetBackup Auto Image Replication" on page 158.
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 165.
Step 3	Add the remote storage server as a replication target	See "Configuring a target for MSDP replication to a remote domain" on page 182. See "Viewing the replication topology for Auto Image Replication" on page 163.
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 189. See "About the storage lifecycle policies required for Auto Image Replication" on page 190. See "Creating a storage lifecycle policy" on page 192.
Step 5	Configure replication bandwidth	Optionally, you can configure the bandwidth for replication. See "About configuring MSDP optimized duplication and replication bandwidth" on page 188.

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 159.
- 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 **Media Server Deduplication Pool** in one NetBackup domain to a **Media Server Deduplication Pool** in another domain.

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 master servers in the source and the target domains so that the master server in the target domain can import the images as soon as they are ready. The master 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 6-25 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 190.

Table 6-25 Auto Image Replication process overview

Event	Domain in which event occurs	Event description
1	Originating master 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 master server (Domain 2)	The storage server in the target domain recognizes that a replication event has occurred. It notifies the NetBackup master server in the target domain.
3	Target master 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 master 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 master server domain, either create multiple source copies or cascade duplication to target master servers.

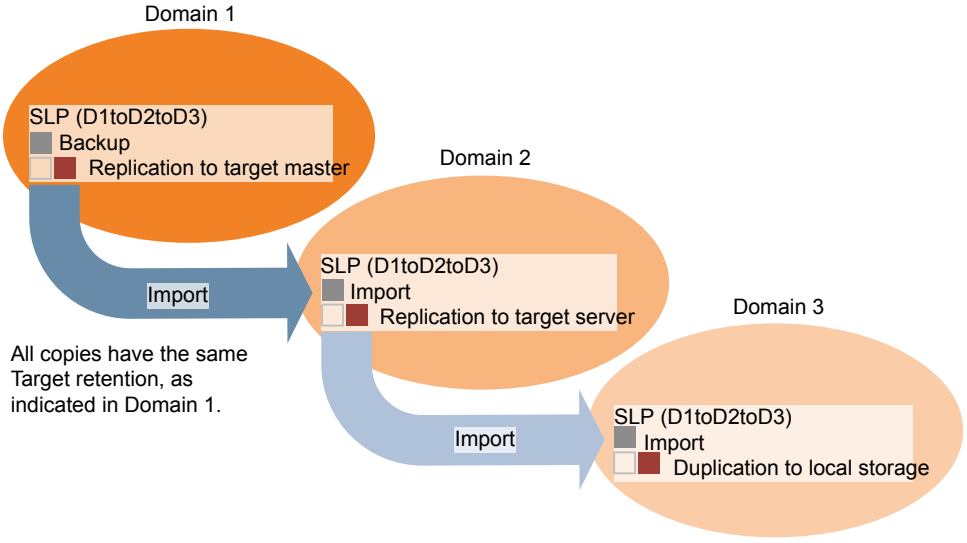
Cascading Auto Image Replication model

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 master.

Figure 6-6 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 6-6 Cascading Auto Image Replication



In the cascading model, the originating master server for Domain 2 and Domain 3 is the master server in Domain 1.

Note: When the image is replicated in Domain 3, the replication notification event indicates that the master server in Domain 2 is the originating master server. However, after the image is imported successfully into Domain 3, NetBackup correctly indicates that the originating master server is in Domain 1.

The cascading model presents a special case for the Import SLP that replicates the imported copy to a target master. (This master server that is neither the first nor the last in the string of target master 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 6-26 shows the difference in the import operation setup.

Table 6-26 Import operation difference in an SLP configured to replicate the imported copy

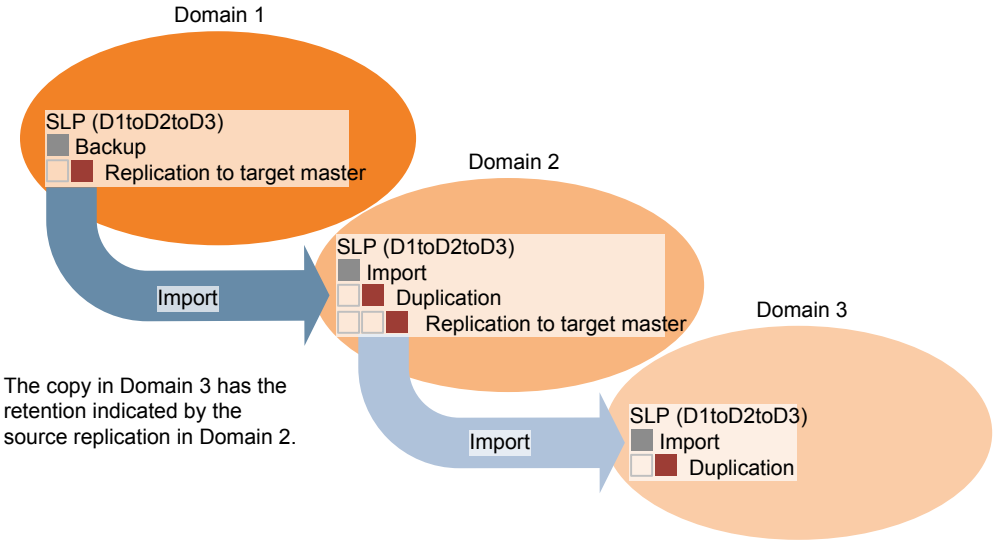
Import operation criteria	Import operation in a cascading model
The first operation must be an import operation.	Same; no difference.
A replication to target master must use a **Fixed** retention type	Same; no difference.
At least one operation must use the **Target retention**.	Here is the difference: 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 6-6, 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 master. 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 6-7 Cascading replications to target master 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 182.

Before you configure the replication relationship, you can add the target master server as a trusted host.

See "About trusted primary servers for Auto Image Replication" on page 165.

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. To do so, you add target storage servers 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 182.

NetBackup discovers the replication topology when you configure the replication relationships. NetBackup discovers topology changes when you use the **Refresh** option of the **Change Disk Pool** dialog box.

See "Changing a Media Server Deduplication Pool properties" on page 437.

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 163.

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 162.

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 164.

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 master 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 Administration Console** does not support adding a trusted primary server using an external CA-signed certificate.

See "Adding a trusted primary server using external CA-signed certificate" on page 176.

See "About the certificate to be used for adding a trusted master server" on page 169.

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 6-8 Tasks to establish a trust relationship between primary servers for targeted A.I.R. using NetBackup CA-signed certificate

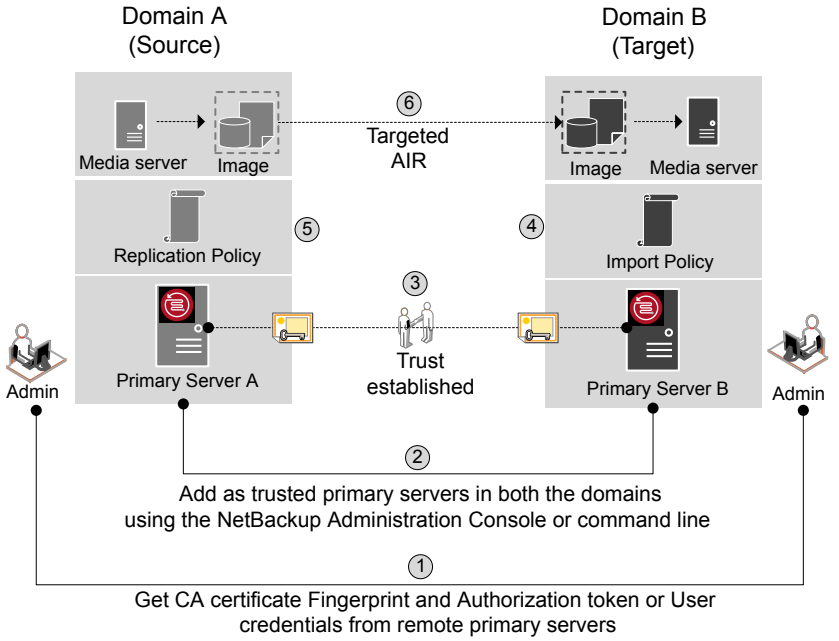


Table 6-27 Tasks to establish a trust relationship between primary servers for targeted A.I.R.

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 6-27 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 Administration Console, see the following topic: See "Adding a trusted master server using a NetBackup CA-signed (host ID-based) certificate" on page 171. 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 180. See "Configuring a NetBackup Deduplication Engine user with limited permissions for Auto Image Replication" on page 187.
Step 4	Create an import storage lifecycle policy in the target domain.	See "About storage lifecycle policies" on page 189.
Step 5	On the source MSDP server, use the **Replication** tab from the **Change Storage Server** dialog box to add the credentials of the target storage server.	See "Configuring a target for MSDP replication to a remote domain" on page 182.
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 189.

Table 6-27 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 158.

If your source and target trusted servers use different NetBackup versions, consider the following.

Note: When you upgrade both the source and the target primary server to version 8.1 or later, you need to update the trust relationship. Run the following command:

```
nbseccmd -setuptrustedmaster -update
```

See the NetBackup Commands Reference Guide.

Table 6-28 Trust setup methods for different NetBackup versions

Source server version	Target server version	Trust setup method
8.1 and later	8.1 and later	Add a trusted primary server using authorization token. Complete action on both the servers.
8.1 and later	8.0 or earlier	On the source server, add the target as the trusted primary server using the remote (target) server's credentials.
8.0 or earlier	8.1 and later	On the source server, add the target as the trusted primary server using the remote (target) server's credentials.

About the certificate to be used for adding a trusted master server

Source or target master servers 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 master servers, NetBackup verifies the following:

Can the source master server establish trust using 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 master server, it can establish the trust using an external certificate. In case of Windows certificate trust store, only `ECA_CERT_PATH` is defined. For more information on the configuration options, refer to the NetBackup Administrator's Guide, Volume I.
Which certificate authorities (CA) does the target master server support?	The target master server may support external CA, NetBackup CA, or both. The following settings show the CA usage information of the master server: ■ In the NetBackup Administration Console - **NetBackup Management** > **Security Management** > **Global Security Settings** ■ In the NetBackup web user interface - **Security > Global Security Settings** > **Secure Communication**.

The following table lists CA support scenarios and certificate to be used to establish trust between the source and the target master servers.

Table 6-29 Certificate to be used for trust setup

Source master server capability to use external certificate	CA usage of the target master server	Certificate to be used for trust setup
Yes The source master server can use NetBackup CA and external CA for communication with a remote master server	External CA	External CA See "Adding a trusted primary server using external CA-signed certificate" on page 176.
	NetBackup CA	NetBackup CA See "Adding a trusted master server using a NetBackup CA-signed (host ID-based) certificate" on page 171.
	External CA and NetBackup CA	NetBackup prompts to select the CA that you want to use for trust setup ■ If you choose to use external CA, do the following: See "Adding a trusted primary server using external CA-signed certificate" on page 176. ■ If you choose to use NetBackup CA, do the following: See "Adding a trusted master server using a NetBackup CA-signed (host ID-based) certificate" on page 171.

Table 6-29 Certificate to be used for trust setup *(continued)*

Source master server capability to use external certificate	CA usage of the target master server	Certificate to be used for trust setup
No The source master server can use only NetBackup CA for communication with a remote maser server	External CA	No trust is established
	NetBackup CA	NetBackup CA See "Adding a trusted master server using a NetBackup CA-signed (host ID-based) certificate" on page 171.
	External CA and NetBackup CA	NetBackup CA See "Adding a trusted master server using a NetBackup CA-signed (host ID-based) certificate" on page 171.

Adding a trusted master server using a NetBackup CA-signed (host ID-based) certificate

Replication operations require that a trust relationship exists between the NetBackup servers in the different domains.

Before you begin

Perform the following steps on both the source and the target server:

- Identify the NetBackup versions that are installed on the source and the target servers.
- Obtain the authorization tokens of the remote server.
  Use the `bpnbat` command to log on and `nbcertcmd` to get the authorization tokens.
- Obtain the fingerprints for the remote server.
  To obtain the SHA1 fingerprint of root certificate, use the `nbcertcmd -displayCACertDetail` command.
- Ensure that you have one of the following permissions:
  - System administrator permissions with `root` permissions for UNIX, administrator permissions for Windows, or a NetBackupCLI user for appliances with software versions 3.1 and later.
  - Access to the NetBackup Administration Console, where you have **<username> ADMIN=ALL** permissions through `auth.conf`.
  - Enhanced Auditing (EA) user permissions through `authalias.conf`.

- For remote Windows master server, if the user's domain is not same as that of the authentication service, you must add the domain with LDAP using the `vssat addldapdomain` command. See the NetBackup Commands Reference Guide.
  Also, this user must have RBAC security administrator permissions. See the NetBackup Web UI Administrator's Guide.

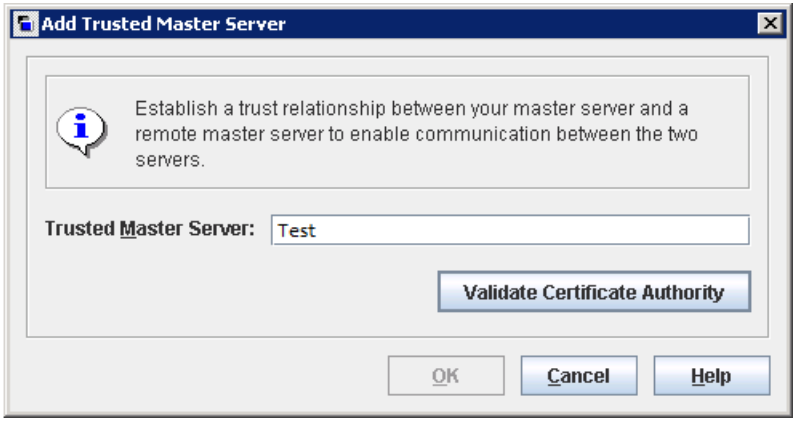
Adding a trusted master server, when both the source and the target servers are NetBackup version 8.1 or later

Use this procedure to add a trusted master server when both the source and target servers are NetBackup version 8.1 or later.

See "Adding a trusted primary server using external CA-signed certificate" on page 176.

To add a trusted master server, when both the source and the target servers are NetBackup version 8.1 or later

1. In the **NetBackup Administration Console**, expand **NetBackup Management > Host Properties > Master Servers** in the left pane.
2. In the right pane, select the master server and **Actions** > **Properties**.
3. In the properties dialog box left pane, select **Servers**.
4. On the **Trusted Master Servers** tab, click **Add**.
5. Enter the fully-qualified host name of the remote master server and click **Validate Certificate Authority**.



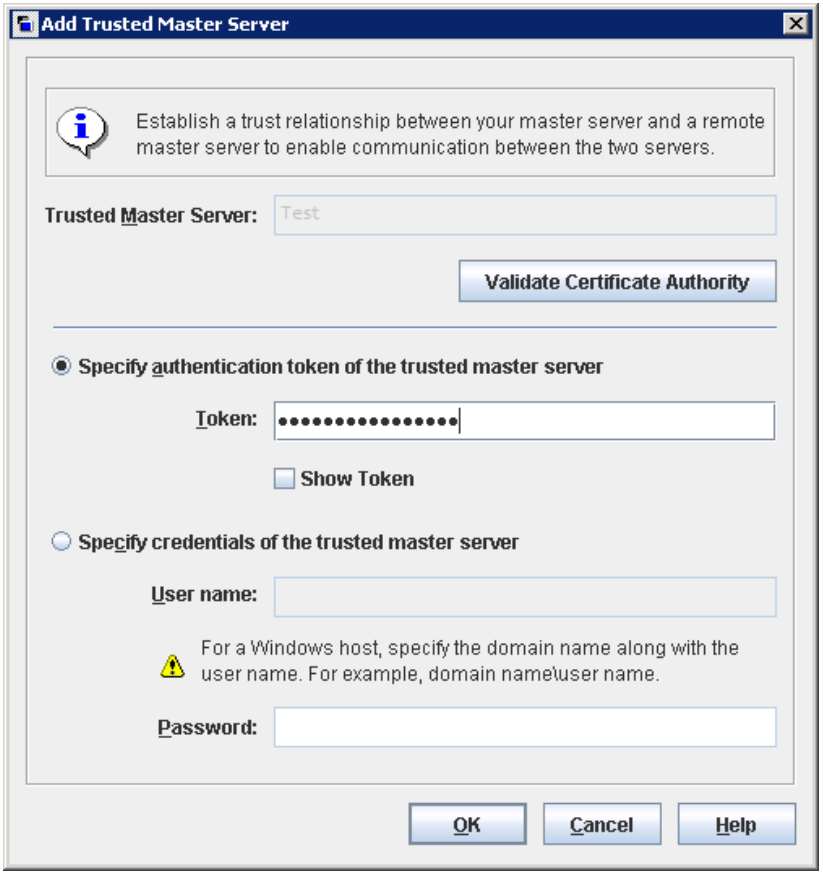
6 In the **Validate Certificate Authority** dialog box, verify if the CA certificate fingerprint of the remote server is correct.

To proceed, click **Yes**.

If the fingerprints don't match, click **No**. Contact the remote server admin to provide the correct fingerprints.

7 Enter the trusted master server details using one of the following methods.

- (Recommended) Select **Specify authentication token of the trusted master server** and enter the token details of the remote master server.
- Select **Specify credentials of the trusted master server** and enter the user name and password. Note that this 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.



8 Click **OK**.

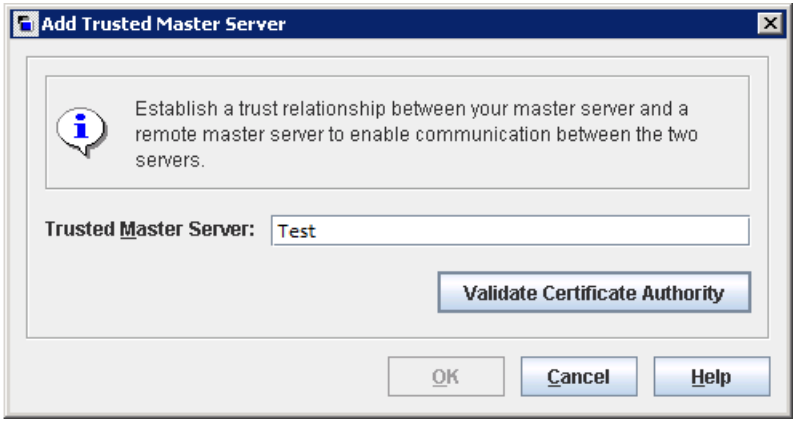
9 Perform the same procedure on the remote master server that you added in step 5.

Adding a trusted master server, when both the source and the target server are NetBackup version 8.0

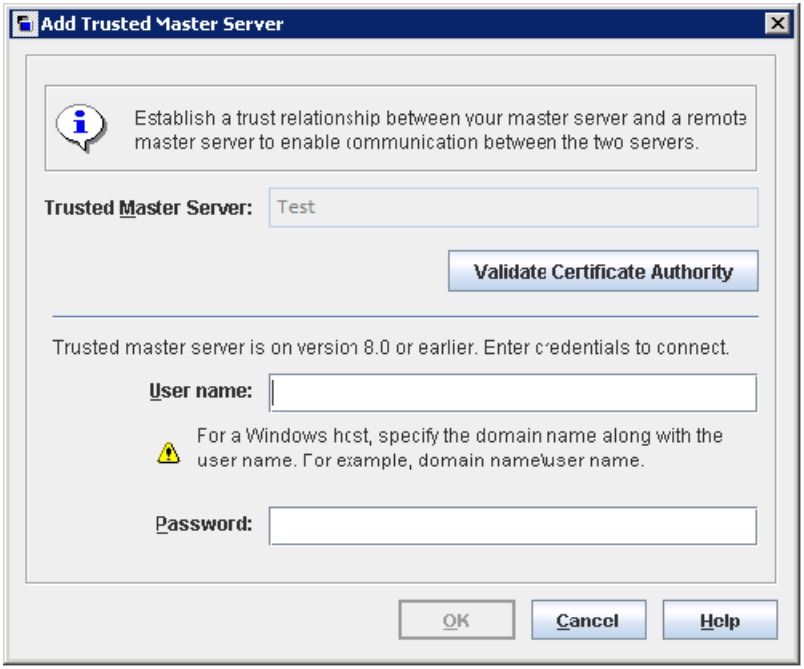
Use this procedure to add a trusted master server when both the source and target servers are NetBackup version 8.0.

To add a trusted master server, when both the source and the target server are NetBackup version 8.0

1. Ensure that the **Enable insecure communication with NetBackup 8.0 and earlier hosts** option is enabled in the global security settings.
2. In the **NetBackup Administration Console**, expand **NetBackup Management > Host Properties > Master Servers** in the left pane.
3. In the right pane, select the master server and **Actions** > **Properties**.
4. In the properties dialog box left pane, select **Servers**.
5. On the **Trusted Master Servers** tab, click **Add**.
6. Enter the fully-qualified host name of the remote master server and click **Validate Certificate Authority**.



7 Enter the **Username** and **Password** of the remote master server host.



8 Click **OK**.

More information

See "About trusted primary servers for Auto Image Replication" on page 165.

See "Configuring MSDP replication to a different NetBackup domain" on page 156.

For details on usage reporting in the web UI, see the *NetBackup Web UI for Administrator's Guide*.

For more information on commands, see the NetBackup Commands Reference Guide. For details on the `authalias.conf`, see the NetBackup Security and Encryption Guide.

Adding a trusted primary server using external CA-signed certificate

You can now establish a trust between source and target primary servers using an external CA-signed certificate.

For more information on the external CA support, refer to the *NetBackup Security and Encryption Guide*.

See "About the certificate to be used for adding a trusted master server" on page 169.

Note: The **NetBackup Administration Console** does not support adding a trusted primary server using an external certificate.

If you try to add a trusted primary server with an external certificate using the **NetBackup Administration Console**, an error is displayed.

To add a trusted primary server using an external certificate

1. Configure the following external certificate configuration options on the source primary server:
   - ECA_CERT_PATH

     Note: In case of Windows certificate store, configure only the `ECA_CERT_PATH` configuration option.

   - ECA_PRIVATE_KEY_PATH
   - ECA_TRUST_STORE_PATH
   - ECA_KEY_PASSPHRASEFILE (optional)

     Note: Do not use the ECA_KEY_PASSPHRASEFILE on the MSDP servers that are used for MSDP direct cloud tiering as it is not supported with MSDP direct cloud tiering.

2. Run the `nbseccmd -setuptrustedmaster` command on the source primary server.

   For more information on the commands, refer to the NetBackup Commands Reference Guide.

If the source and target primary servers are configured with external certificates issued by different certificate authorities, refer to the following section from the *NetBackup Deduplication Guide*: *Configuring external CA for secure communication between the source MSDP storage server and the target MSDP storage server*

Removing a trusted primary server

To remove a trusted primary server, you must perform the following procedure on both the source and the target server.

Note: If either your source or the target server is on version 8.0 or earlier, follow the procedure that is prescribed in the respective guide.

To remove a trusted primary server

1. Ensure that all replication jobs to the trusted target primary server are complete. You can use `nbstlutil stlilist` to list the state of all storage lifecycle policy-managed operations. To cancel jobs use `nbstlutil cancel`.

   See the NetBackup Commands Reference Guide for information about the `nbstlutil` command.

2. Delete all storage lifecycle policies (SLPs) that use the trusted primary as a destination.

   Note: Before deleting a storage lifecycle policy, ensure that there are no backup policies that indicate the SLP for the **Policy storage**.

3. In the **NetBackup Administration Console**, expand **NetBackup Management > Host Properties > Primary Servers** in the left pane.
4. In the right pane, select the primary server.
5. On the **Actions** menu, click **Properties**.
6. In the properties dialog box left pane, select **Servers**.
7. In the **Servers** dialog box, select the **Trusted Primary Servers** tab.
8. On the **Trusted Primary Servers** tab, select the trusted primary server that you want to remove and click **Remove**.

   The **Remove Server** confirmation dialog box is displayed.
9. Click **Yes**.
10. When you finish removing trusted primary servers, click **OK**.
11. Restart the `nbsl` service.
12. Repeat the steps on the source primary server.

Note: In case of multiple NICs, if you have 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.

Enabling NetBackup clustered primary server inter-node authentication

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. The inter-node authentication allows the following NetBackup functionality:

NetBackup Administration Console	The NetBackup Administration Console 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. See "About trusted primary servers for Auto Image Replication" on page 165.

To enable clustered primary server inter-node authentication

- 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 master 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 master server:

   - Windows

     install_path\NetBackup\bin\nbcertcmd -getCACertificate -server *target_master_server*

   - UNIX

     /usr/openv/netbackup/bin/nbcertcmd -getCACertificate -server *target_master_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 master server:
   - Windows
     install_path\NetBackup\bin\nbcertcmd -getCertificate -server *target_master_server* -token *token_string*
   - UNIX
     /usr/openv/netbackup/bin/nbcertcmd -getCertificate -server *target_master_server* -token *token_string*
4. Use either of these two methods to obtain the authorization tokens:
   - NetBackup Administration Console
     - Log on the target NetBackup master server and open **Security Management** > **Certificate Management** > **Token Management**.
     - Click the **Create Token** option to create a token, or right-click the blank area of the **Token records** list view and select the **New Token** menu item to create a token.
   - NetBackup Commands
     - Use the `bpnbat` command to log on the target NetBackup master 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 155.

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 156.

Note: About clustered master servers: If you add a trusted master 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 mastered server is the source of the replication operation or the target.

See "About trusted primary servers for Auto Image Replication" on page 165.

See "Enabling NetBackup clustered primary server inter-node authentication" on page 179.

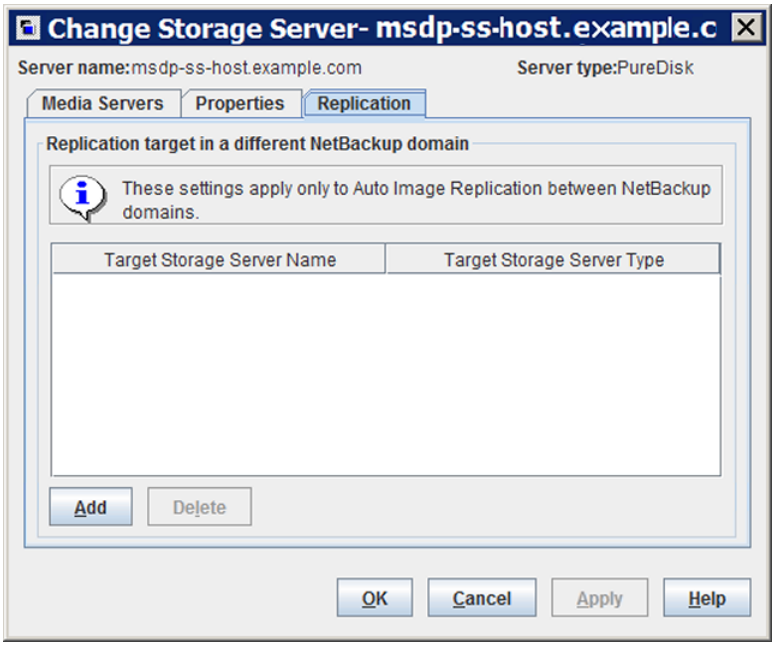
Caution: Choose the target storage server or servers carefully. A target storage server must not also be a storage server for the source domain. Also, a disk volume must not be shared among multiple NetBackup domains.

To configure a Media Server Deduplication Pool as a replication target

1. In the **NetBackup Administration Console**, expand **Media and Device Management > Credentials > Storage Server**.
2. Select the MSDP storage server.
3. On the **Edit** menu, select **Change**.

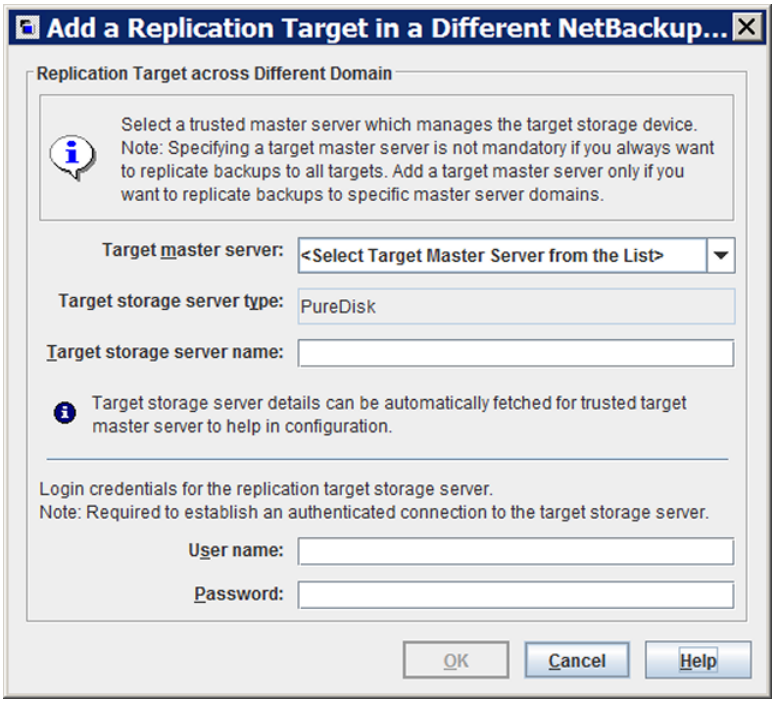
4 In the **Change Storage Server** dialog box, select the **Replication** tab.

The following is an example of the **Change Storage Server** dialog box **Replication** tab:



5 On the **Replication** tab, click **Add**. The **Add a Replication Target Across a Different NetBackup Domain** dialog box appears.

The following is an example of the dialog box.



6 In the **Add a Replication Target Across a Different NetBackup Domain** dialog box, complete one or more of the following procedures depending on your requirements:

Add a trusted master server

Add a trusted master server if you want to replicate backup images to a subset of available targets.

To add a trusted master server

1. In the **Target master server** drop-down list, select **Add a new trusted master server**.
2. Complete the fields in the **Add a New Trusted Master Server** dialog box. Click **OK** after you complete the fields.

   See "Target options for MSDP replication" on page 186.
3. Repeat the first two steps until you have finished adding trusted master servers.
4. To add a replication target, continue with the next procedure.

See "About trusted primary servers for Auto Image Replication" on page 165.

Select a trusted master server and replication target

Select a trusted master server if you want to replicate backup images to a subset of available targets.

To select a trusted master server and replication target

1. In the **Target master server** drop-down list, select the master server of the domain to which you want to replicate data.

   All trusted master servers are in the drop-down list.
2. In the **Target storage server type** drop-down list, select the type of target storage server.

   All available target types are in the drop-down list.
3. In the **Target storage server name** drop-down list, select the storage server that hosts the target deduplication pool. All available storage servers in the target domain are in the drop-down list.

   After you select the target storage server, NetBackup populates the **User name** field with the **User name** of the NetBackup Deduplication Engine of the target host.
4. Enter the **Password** for the deduplication service on the target storage server.
5. Click **OK**.

See "Target options for MSDP replication" on page 186.

Enter a replication target

Enter a replication target if you did not configure trusted master servers.

To enter a replication target

1. In the **Target storage server name** field, enter the name of the target storage server.
2. Enter the **User name** and **Password** for the NetBackup Deduplication Engine on the target storage server.
3. Click **OK**.

See "Target options for MSDP replication" on page 186.

7. After all replication targets are added, click **OK**.
8. For the deduplication pools in each domain, open the **Change Disk Pool** dialog box and click **Refresh**.

   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 437.

Target options for MSDP replication

The following table describes the target options for replication to a NetBackup **Media Server Deduplication Pool**.

Table 6-30 MSDP target replication options

Option	Description
Target master server	All trusted master servers are in the drop-down list. Select the master server for the target domain to which you want to replicate backups. To add the master server of another domain as a trusted master, 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 master server is configured, the value is **Target storage server name**. If a trusted master server is not configured, the value is **PureDisk**.

Table 6-30 MSDP target replication options *(continued)*

Option	Description
Target storage server name	If a trusted master server is configured, select the target storage server. If a trusted master 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 187.
Password	Enter the password for the NetBackup Deduplication Engine.

See "Configuring a target for MSDP replication to a remote domain" on page 182.

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 master server, enter the user name and password of the user with limited permissions for AIR.

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`

OPTDUP_BANDWIDTH	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 208. See "Editing the MSDP pd.conf file" on page 209. See "MSDP pd.conf file parameters" on page 209.

See "Configuring MSDP optimized duplication within the same NetBackup domain" on page 148.

See "Configuring MSDP replication to a different NetBackup domain" on page 156.

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

Note: SLPs can be configured from the NetBackup web UI. To view the existing SLPs or create a new one, on the left navigation pane, click **Storage > 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.

An SLP 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 Parameters** properties in the **NetBackup Administration Console** allow administrators to customize how SLPs are maintained and how SLP jobs run.

Best-practice information about SLPs appears in the following document:

https://www.veritas.com/content/support/en_US/article.100009913

For more information, see the NetBackup Administrator's Guide, Volume I.

About the storage lifecycle policies required for Auto Image Replication

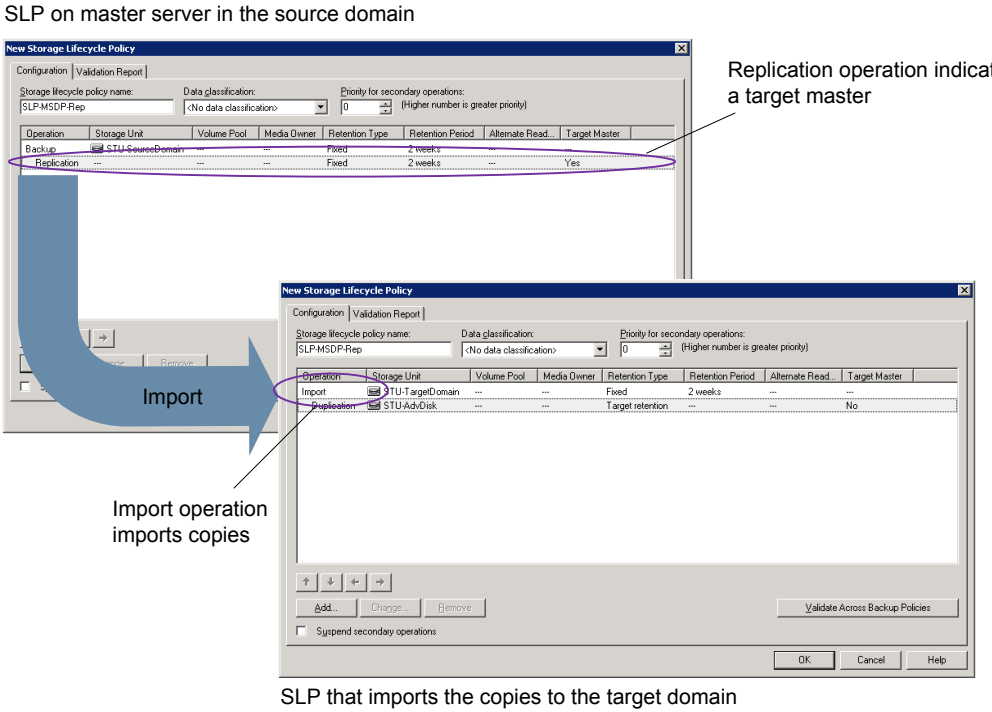
To replicate images from one NetBackup domain to another NetBackup domain requires two storage lifecycle policies. The following table describes the policies and their requirements:

Table 6-31 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 master server domains or only to specific replication targets. See "About trusted primary servers for Auto Image Replication" on page 165. ■ The SLP must be of the same data classification as the **Import** SLP in Domain 2.
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. See the following topic for more information about **Replication** operation configuration:

Figure 6-9 shows how the SLP in the target domain is set up to replicate the images from the originating master server domain.

Figure 6-9 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.

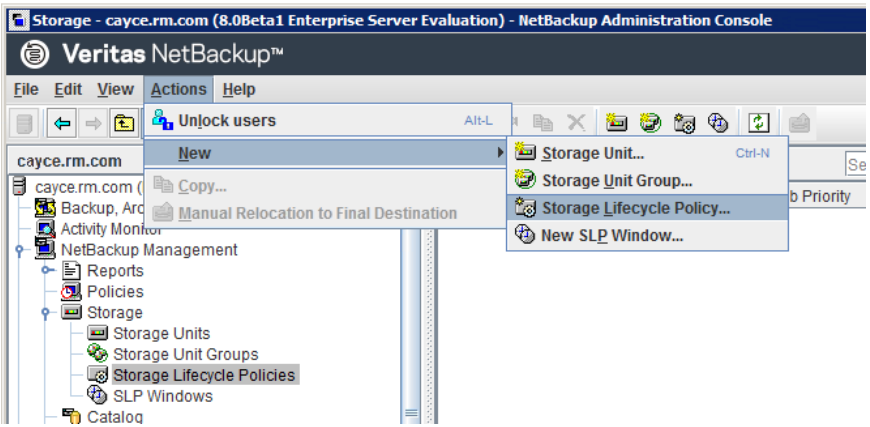
Creating a storage lifecycle policy

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.

Note: You can create an SLP and add multiple storage operations to it from the NetBackup web UI. To add an SLP, on the left navigation pane, click **Storage > Storage Lifecycle Policies**, and then click **+Add** button.

To add a storage operation to a storage lifecycle policy

1. In the **NetBackup Administration Console**, select **NetBackup Management > Storage > Storage Lifecycle Policies**.
2. Click **Actions > New > Storage Lifecycle Policy**.



3. In the **New Storage Lifecycle Policy** dialog box, enter a **Storage lifecycle policy name**.

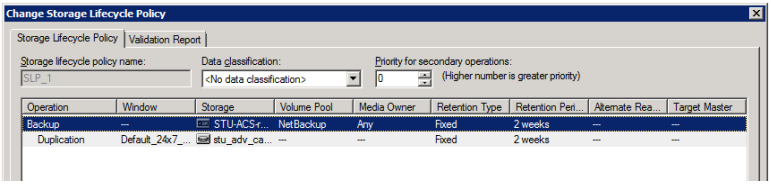
   See “NetBackup naming conventions” on page 31.

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.

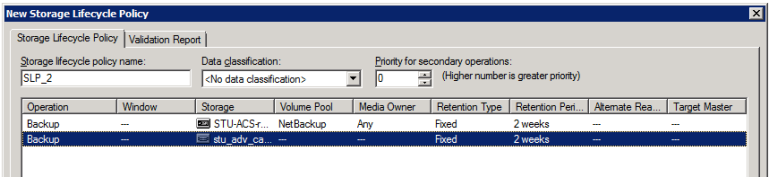
If this is the first operation added to the SLP, click **Add**.

If this is not the first operation in the SLP, add an operation that is either hierarchical or non-hierarchical:

To create a hierarchical operation, select an operation to become the source of the next operation. Click **Add**. The new operation is a child of the selected operation. The child is indented under the parent operation.



To create a non-hierarchical operation, do not select an operation. A non-hierarchical operation means that the operation does not have a parent and child relationship with another operation. The new operation is not indented.



5 In the **Properties** tab of the **New Storage Operation** dialog box, 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've selected.

The name of the operation reflects its purpose in the SLP:

- **Backup**
- **Duplication**
- **Import**
- **Replication**
  See “About NetBackup Auto Image Replication” on page 158.

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 Click the **Advanced** button in the **Properties** tab to display options about how the window should behave if the window closes and a secondary operation is not yet complete.

9 Click **OK** 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 Click **OK** 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**.

See "Creating a backup policy" on page 199.

Storage Lifecycle Policy dialog box settings

The **New Storage Lifecycle Policy** dialog box and the **Change Storage Lifecycle Policy** dialog box contain the following settings.

Note: The SLP options can be configured on the NetBackup web UI.

Figure 6-10 Storage Lifecycle Policy tab

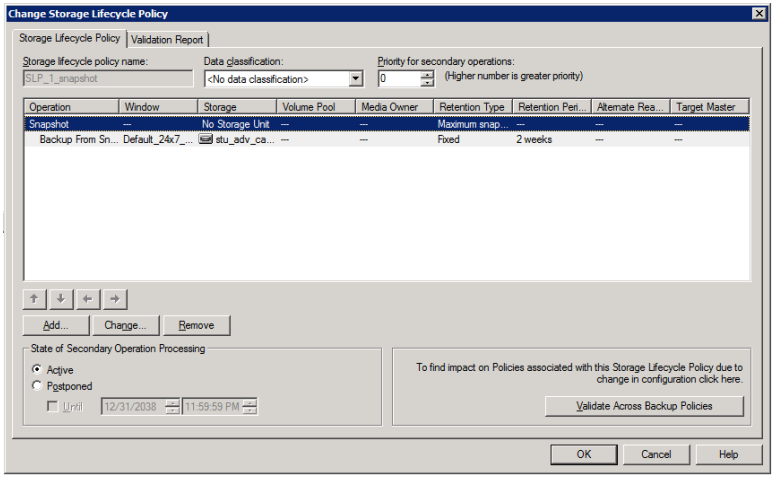


Table 6-32 Storage Lifecycle Policy tab

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 6-32 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 drop-down 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. It 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 190. 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 6-32 Storage Lifecycle Policy tab *(continued)*

Setting	Description
Operations	Use the **Add**, **Change**, and **Remove** buttons 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 display by default. For column descriptions, see the following topic:
Arrows	Use the arrows to indicate the indentation (or hierarchy) of the source for each copy. One copy can be the source for many other copies.
Active and **Postponed**	The **Active** and **Postponed** options appear under **State of Secondary Operation Processing** and refer to the processing of all duplication operations in the SLP. **Note:** The **Active** and **Postponed** options apply to duplication operations that create tar-formatted images. For example, those created with `bpduplicate`. The **Active** and **Postponed** options do not affect the images that are duplicated as a result of OpenStorage optimized duplication, NDMP, or 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 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.
Validate Across Backup Policies button	Click this button to see how changes to this SLP can affect the policies that are associated with this SLP. The button generates a report that displays on the **Validation Report** tab. This button 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.

Creating a backup policy

Use the following procedure to create a backup policy.

To create a policy

1. In the **NetBackup Administration Console**, expand **NetBackup Management > Policies**.
2. Select **Actions** > **New > Policy**.
3. Type a unique name for the policy.
4. Clear the **Use Policy Configuration Wizard** and click **OK**.
5. Configure the attributes, the schedules, the clients, and the backup selections for the new policy.

Resilient Network properties

Note: The **Resilient Network** properties can be configured on the NetBackup web UI. On the left pane, select **Hosts > Host properties**. Depending on the host to be configured, select the **Primary Server**, **Media Servers**, or **Clients**. From the Actions menu, select the **Edit** option, and then click **Resilient Network**.

The **Resilient Network** properties appear for the primary server, for media servers, and for clients. 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 different subdomain than the server, add the fully qualified domain name of the server to the client's hosts file. For example, `india.veritas.org` is a different subdomain than `china.veritas.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.

Figure 6-11 Primary server Resilient Network host properties

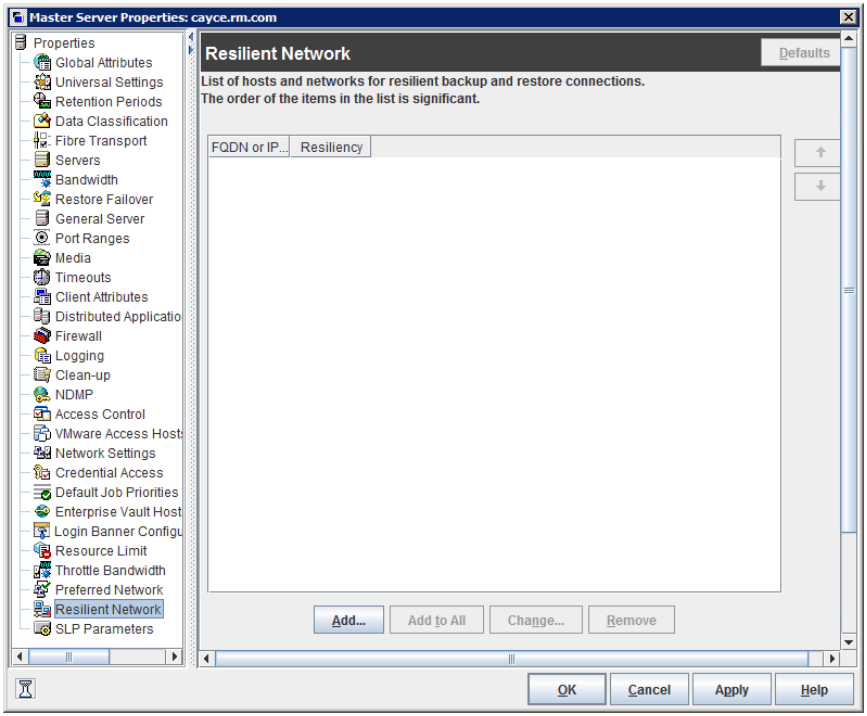


Table 6-33 describes the **Resilient Network** properties.

Table 6-33 Resilient Network dialog box properties

Property	Description
Host Name or IP Address	The **Host 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.

The resilient status of each client also appears as follows:

- In the **NetBackup Administration Console**, select **NetBackup Management > Policies** in the left pane and then select a policy. In the right pane, a **Resiliency** column shows the status for each client in the policy.
- In the **NetBackup Administration Console**, select **NetBackup Management > Host Properties > Clients** in the left pane. In the right pane, a **Resiliency** column shows the status for each client.

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.

See the NetBackup Logging Reference 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.

Specifying resilient connections

Use the following procedure to specify resilient connections for NetBackup clients.

See "Resilient Network properties" on page 199.

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

1. In the **NetBackup Administration Console**, expand **NetBackup Management > Host Properties > Primary Servers** in the left pane.
2. In the right pane, select the primary server on which to specify properties.
3. On the **Actions** menu, click **Properties**.
4. In the properties dialog box left pane, select **Resilient Network**.
5. In the **Resilient Network** dialog box, use the following buttons to manage resiliency for clients:

Add

To add resilient settings

1. Click **Add**.

   The **Add Resilient Network Settings** dialog box appears

2. Enter a client host name, an IP address, or an address range.

   If you specify the client host by name, it is recommended that you use the fully qualified domain name.

   For address ranges, use Classless Inter-Domain Routing (CIDR) notation (for example, **192.168.100.0/24** or **fd00::/8**).

3. Ensure that the **Resiliency On** option is selected.
4. Click **Add**.
5. Repeat until you have finished entering clients or address ranges.
6. When you finish adding network settings, click **Close**.

Add To All

If you select multiple hosts in the **NetBackup Administration Console**, the entries in the **Resilient Network** list may appear in different colors, as follows:

- The entries that appear in black type are configured on all of the hosts.
- The entries that appear in gray type are configured on some of the hosts only.

For the entries that are configured on some of the hosts only, you can add them to all of the hosts. To do so, select them and click **Add To All**.

Change

To change resilient settings

1. Select the client host name, the IP address, or the address range.
2. Click **Change**.

   The **Change Resilient Network Settings** dialog box appears

3. Select the desired **Resiliency** setting.
4. Click **OK**.

Remove

Remove the select host or address rang

1. Select the client host name, the IP address, or the address range.
2. Click **Remove**.

   The client is removed immediately; a confirmation dialog box does not appear.

Move an item in the list of items

1. Select the client host name, the IP address, or the address range.
2. Click the appropriate button to move up the item or move down the item.

   The order of the items in the list is significant.

   See "Resilient Network properties" on page 199.

6. After specifying resilient connections, click **OK**.

   The settings are propagated to the affected hosts through normal NetBackup inter-host communication, which can take up to 15 minutes.

7. If you want to begin a backup immediately, restart the NetBackup services on the primary server.

Adding an MSDP load balancing server

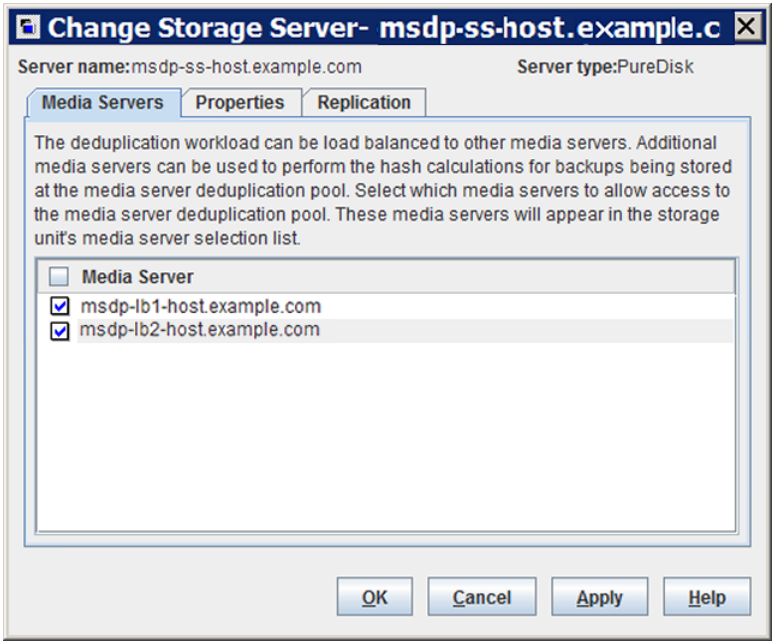
You can add a load balancing server to an existing media server deduplication node.

See "About MSDP storage servers" on page 39.

To add a load balancing server

1. In the NetBackup Administration Console, expand **Media and Device Management > Credentials > Storage Server**
2. Select the deduplication storage server.

3 On the **Edit**, select **Change**.



4 In the **Change Storage Server** dialog box, select the **Media Servers** tab

5 Select the media server or servers that you want to use as a load balancing server. It must be a supported host.

The media servers that are checked are configured as load balancing servers.

6 Click **OK**.

7 For all storage units in which **Only use the following media servers** is configured, ensure that the new load balancing server is selected.

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 6-34 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.

Table 6-34 Effect of variable-length deduplication *(continued)*

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. You can enable variable-length deduplication 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.

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 "About the MSDP pd.conf configuration file" on page 208.

See "Editing the MSDP pd.conf file" on page 209.

See "MSDP pd.conf file parameters" on page 209.

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 209.

See "Editing the MSDP pd.conf file" on page 209.

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: Veritas recommends that you make a backup copy of the file before you edit it.

See "About the MSDP pd.conf configuration file" on page 208.

See "MSDP pd.conf file parameters" on page 209.

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 6-35 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 6-35 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 127.

Table 6-35 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, Veritas 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 6-35 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 6-35 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 129. To have all data in MSDP server encrypted, it is recommended that you use the server option. `pd.conf` parameter impacts only the backups or replication using this hosts as the data source.
FIBRECHANNEL	Enable Fibre Channel for backup and restore traffic to and from a NetBackup series appliance. Default value: `FIBRECHANNEL = 0` Possible values: `0` (off) or `1` (on)

Table 6-35 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 199. ■ 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 6-35 pd.conf file parameters *(continued)*

Parameter	Description
FP_CACHE_CLIENT_POLICY	**Note:** Veritas 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 85.
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 Veritas 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 6-35 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 Veritas 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 Veritas 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 453.
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 453.

Table 6-35 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 Veritas 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 Veritas 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 Veritas representative.

Table 6-35 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 75. 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 72.

Table 6-35 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 72.

Table 6-35 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 75. 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 72.
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 72.
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 188.

Table 6-35 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 127.
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 129.
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 6-35 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 Veritas 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 Veritas representative. You can also specify the segment size for specific file types. See PREFERRED_EXT_SEGKSIZE.

Table 6-35 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 6-35 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 208.

See “Editing the MSDP pd.conf file” on page 209.

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 Veritas 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 Veritas representative.

The `contentrouter.cfg` file resides in the following directories:

- (UNIX) *storage_path*`/etc/puredisk`
- (Windows) *storage_path*`\etc\puredisk`

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, Veritas 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 226.

See "Editing an MSDP storage server configuration file" on page 227.

See "Setting the MSDP storage server configuration" on page 228.

Saving the MSDP storage server configuration

Veritas 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 225.

See "Recovering from an MSDP storage server disk failure" on page 465.

See "Recovering from an MSDP storage server failure" on page 466.

To save the storage server configuration

- On the master 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 6-36 shows the configuration lines that are required.

Table 6-36 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.

Table 6-36 Required lines for a recovery file *(continued)*

Configuration setting	Description
`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.
`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 master 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 225.

See "Recovering from an MSDP storage server disk failure" on page 465.

See "Recovering from an MSDP storage server failure" on page 466.

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 226.

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 6-36) 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 465.

See "Recovering from an MSDP storage server failure" on page 466.

To set the configuration, you must have an edited storage server configuration file.

See "About saving the MSDP storage server configuration" on page 225.

See "Saving the MSDP storage server configuration" on page 226.

See "Editing an MSDP storage server configuration file" on page 227.

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 master server, run the following command:

  UNIX: `/usr/openv/netbackup/bin/admincmd/nbdevconfig -setconfig -storage_server` *`sshostname`* `-stype PureDisk -configlist` *`file.txt`*

  Windows: *`install_path`*`\NetBackup\bin\admincmd\nbdevconfig -setconfig -storage_server` *`sshostname`* `-stype PureDisk -configlist` *`file.txt`*

  For *sshostname*, use the name of the storage server. For *file.txt*, use the name of the file that contains the configuration.

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`

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 229.

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 427.

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 231.
Catalog backup policy	Veritas 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 232.

See "About recovering the MSDP catalog" on page 462.

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. Veritas 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 234.

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 235.

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 236.

About storing MSDP catalog shadow copy duplicates on data volumes

MSDP lets you store two additional duplicates of the catalog shadow copies on separate data volumes. The storage of the duplicates allows for better metadata resiliency.

You must use the `cacontrol` command to add or delete a volume to store the duplicate. When you use `cacontrol`, any changes to the configuration take effect during the next catalog backup or recovery and the duplicates are then updated in parallel with the primary duplicate.

The following are examples of how to use the `cacontrol` command:

- To add an additional volume, run:

  ```
  cacontrol --catalog addshadowcopy <data_volume>
  ```

  For example:

  ```
  cacontrol --catalog addshadowcopy /msdp/data/dp1/example_volume1
  ```

- To remove a volume, run:

  ```
  cacontrol --catalog deleteshadowcopy <data_volume>
  ```

  For example:

  ```
  cacontrol --catalog deleteshadowcopy /msdp/data/dp1/example_volume1
  ```

- To see the list of volumes where additional catalog shadow copy duplicates are stored, run:

  ```
  cacontrol --catalog listshadowcopies
  ```

During catalog recovery, MSDP automatically uses the first non-corrupted catalog shadow duplicate it finds if the primary duplicate is corrupted.

About the MSDP catalog backup policy

Veritas 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.con *install_path*\Veritas\NetBackup\bin\ost-plugins\mtstrm *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 237.

See "Updating an MSDP catalog backup policy" on page 241.

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 231.

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 231.

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 * * *
   T  T T T T
   |  | | | |
   |  | | | |
   |  | | | └──── 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 231.

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 58.

See "Troubleshooting MSDP catalog backup" on page 587.

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 master server. See **NetBackup Management > Host Properties > *masterserver_name* > Servers > Additional Servers** in the NetBackup Administration Console.

   If the storage server is not in the **Additional Servers** list, add the MSDP storage server host to the **Additional Servers** 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 238.

   The utility creates a log file and displays its path in the command output.

   See "NetBackup MSDP log files" on page 567.

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 567.

Table 6-37 describes the options for creating and updating an MSDP catalog backup policy.

Table 6-37 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 Veritas 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 6-37 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 6-37 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. Veritas 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 237.

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 58.

See "Troubleshooting MSDP catalog backup" on page 587.

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 238.

  The utility creates a log file and displays its path in the command output.

  See "NetBackup MSDP log files" on page 567.

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
  ```

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
  ```

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_master_server
  install_path\NetBackup\bin\nbcertcmd -getCertificate –server
  another_master_server -token token_string
  ```

- UNIX

  ```
  /usr/openv/netbackup/bin/nbcertcmd -getCACertificate –server
  another_master_server
  /usr/openv/netbackup/bin/nbcertcmd -getCertificate –server
  another_master_server -token token_string
  ```

Use either of these two methods to obtain the authorization tokens:

- NetBackup Administration Console

- Log on the target NetBackup master server and open **Security Management > Certificate Management > Token Management**.
- Click **Create Token** or right-click in the blank area of the **Token records** list view and select **New Token** to create a token.

- NetBackup Commands
  - Use the `bpnbat` command to log on the target NetBackup master 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
`masterA`	`masterB`
`mediaA1`	`mediaB`
`mediaA2`	
`clientA`	

`MasterA` is the host name of the master server of NetBackup domain A and the domain contains two media servers (`mediaA1`, `mediaA2`), and one client (`clientA`). `MasterB` is the host name of the master 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 media server `mediaB` of NetBackup domain B. (NetBackup Administration Console > **Media and Device Management** > **Configure Disk storage servers** > **Media Server Deduplication Pool**)
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 `masterB`.

   ```
   nbcertcmd -GetCACertificate -server masterB
   ```

```
nbcertcmd -GetCertificate -server masterB -token <token_string>
```

4. Create an MSDP OpenStorage server on `mediaA1` of NetBackup domain A. (NetBackup Administration Console > **Media and Device Management** > **Configure Disk storage servers** > **OpenStorage**).

   OpenStorage server type is **PureDisk**, Storage server name is `mediaB`, user name is `testuser1`, and password is `testuser1pass`.

   You must enter the server type as **PureDisk**.

   Now `mediaA1` of 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 masterB`
   - `nbcertcmd -GetCertificate -server masterB -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 masterB`
   - `nbcertcmd -GetCertificate -server masterB -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 optimized duplication and multi-domain are used together, 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 master server of the local NetBackup domain. Run the `nbcertcmd` commands on source side MSDP storage server to request a certificate from the NetBackup master server of 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 master 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.

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 master servers from other domains to connect with the MSDP storage server and the master server of the NetBackup domain that contains the MSDP server. The master 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
`masterA - (10.XX.30.1/24)`	`masterB - (10.XX.40.3/24)`
`masterA2 - (10.XX.40.1/24)`	`mediaB - (10.XX.40.4/24)`
`mediaA - (10.XX.30.2/24)`	
`mediaA2 - (10.XX.40.2/24)`	

`masterA` is the master 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. (NetBackup Administration Console > **Media and Device Management** > **Configure Disk storage servers** > **Media Server Deduplication Pool**)
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 `masterA2` is used as the master server host name of domain A.

   Run following command on `mediaB` to get a CA certificate and a host certificate from `masterA`:

   ```
   nbcertcmd -GetCACertificate -server masterA2
   ```

   ```
   nbcertcmd -GetCertificate -server masterA2 -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 master server:

https://www.veritas.com/support/en_US/article.100034092

4. Create an MSDP OpenStorage server on `mediaB` of NetBackup domain B. NetBackup Administration Console > **Media and Device Management** > **Configure Disk storage servers** > **OpenStorage**).

   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.*

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 *Veritas 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 *Veritas 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 252.

See "About the NetBackup Deduplication Shell" on page 522.

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 251.

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 6-38 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 6-39 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"`

Chapter 7

MSDP cloud support

This chapter includes the following topics:

- About MSDP cloud support
- Create a Media Server Deduplication Pool (MSDP) storage server in the NetBackup web UI
- 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 MSDP cloud immutable (WORM) storage support
- About instant access for object storage in cloud

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

Cloud LSUs can be configured on the storage servers running on Red Hat Linux Enterprise or CentOS platforms. No platform limitations for clients and load balancing servers.

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.
- Cloud DR for cloud LSU of AWS Glacier, AWS Glacier, AWS Deep Archive, and Microsoft Azure Archive is not supported if the storage server name changes.
- The Cloud LSU for AWS Glacier, AWS Deep Archive, and Microsoft Azure Archive cannot be used as either sources or targets 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.
- Image verification for backups residing on a cloud LSU for AWS Glacier, AWS Deep Archive, and Microsoft Azure Archive is not supported.
- SAP HANA for cloud LSU of Microsoft Azure Archive is not supported.
- Snowball 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.

Create a Media Server Deduplication Pool (MSDP) storage server in the NetBackup web UI

Use this procedure to create a Media Server Deduplication Pool (MSDP) storage server in the NetBackup web UI. You have the option to create a disk pool (local storage or cloud storage) and 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. Log into the NetBackup web UI.
2. On the left, click **Storage** > **Storage configuration**and then click **Add**.
3. Select **Media Server Deduplication Pool (MSDP)** from the list.
4. In **Basic properties**, enter all required information and click **Next**.

   You must select your media server by clicking on the field. If you do not see the media server you want to use, you can use **Search** to find it.
5. In **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.
6. (Optional) In **Media servers**, click **Add** to add any additional media servers you want to use.

   Click **Next** after selecting additional media servers or if you want to continue without selecting additional media servers.
7. 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 (drop-down in **Volumes** step) to select an existing disk pool volume or create a new one.

8 (Optional) At the top, click on **Create disk pool**.

9 (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.

Note: Currently, AWS S3 and Azure storage API types are supported.

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: NetBackup Recovery Vault supports multiple options such as Microsoft Azure and Amazon. Contact your Veritas NetBackup account manager for credentials or with any questions about the available options.

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

Creating a cloud storage unit

Use the NetBackup Web UI or the command line to create a cloud storage unit.

To create a cloud storage unit by using the NetBackup Web UI, See “Configuring the MSDP node cloud tier” on page 21.

The following steps describe the method to create a cloud storage unit using the command line:

1. Create an MSDP storage server.

   See "Configuring MSDP server-side deduplication" on page 69.

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 an 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 "lsuCloudUser" " " string`	Specifies the cloud user name.
`V7.5 "lsuCloudPassword" " " string`	Specifies the cloud password.
`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. 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 1: Configuration file with encryption disabled

```
V7.5 "operation" "add-lsu-cloud" string
V7.5 "lsuName" "s3amazon1" string
V7.5 "lsuCloudUser" "CUCU" string
V7.5 "lsuCloudPassword" "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 "lsuCloudUser" "CUCU" string
V7.5 "lsuCloudPassword" "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.

For more information, See "About MSDP Encryption using NetBackup KMS service" on page 91.

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 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 can from the NetBackup Web UI or NetBackup Administration Console.

6 Create 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 or NetBackup Administration Console.

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 "lsuCloudUser" " " string`	Specifies the cloud user name.
`V7.5 "lsuCloudPassword" " " string`	Specifies the cloud password.

For example:

```
V7.5 "operation" "update-lsu-cloud" string
V7.5 "lsuName" "s3amazon1" string
V7.5 "lsuCloudUser" "ChangedCloudUser" string
V7.5 "lsuCloudPassword" "changedpassword" 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>
```

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 KMS service" on page 91.

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:

1. Expire all images of the cloud LSU in NetBackup.
2. Remove the storage unit and disk pool of this MSDP cloud LSU.
3. 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"
4. Run `CRQP` to make sure no `tlog` entries are present in *<msdp_storage_path>*`/spool` folder and *<msdp_storage_path>*`/queue` folder.

5 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>
```

6 Stop the MSDP service and its monitor service.

```
# /usr/openv/netbackup/bin/nbsvcmon -terminate
```

```
# /usr/openv/pdde/pdconfigure/pdde stop
```

7 Delete LSU configurations in `spoold` using the following command:

```
# /usr/openv/pdde/pdcr/bin/spoold --removepartition <storageId>
```

8 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>
```

9 Remove the entire sub-bucket folder in cloud. (Optional)

10 Start the MSDP service and its monitor service.

```
# /usr/openv/pdde/pdconfigure/pdde start
```

```
# /usr/openv/netbackup/bin/nbsvcmon
```

11 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 156.
- Configure a trust relationship with the target NetBackup domain.
  See “About trusted primary servers for Auto Image Replication” on page 165.
- 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 189.

See “About the storage lifecycle policies required for Auto Image Replication” on page 190.
See “Creating a storage lifecycle policy” on page 192.

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 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 user name with the cloud LSU name when you add the A.I.R target. Use Java GUI to add A.I.R. target. The format of the user name with the target cloud LSU is:

```
<username>?LSU=<target cloud LSU>
```

For example, there is a target storage server and user name of the server is `userA` and there is a cloud LSU `s3cloud1` in the target storage server. To replicate image from an old storage server to the cloud LSU of the target server, you can use the following user name while adding the A.I.R target:

```
userA?LSU=s3cloud1
```

You must also create an import SLP to local volume of target storage server in the target master server. Then select the imported SLP while creating the target A.I.R SLP on the source side. When A.I.R runs, the import job in 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 doing a client direct backup 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, new load balance server is selected automatically to make sure that the job can be done successfully. When you are restoring a backup image on cloud LSU and you select the media server explicitly, the media server selected 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
MetaDownloadConnNum	Maximum number of concurrent connections to the cloud provider for downloading metadata. Increasing this value is helpful especially for high latency network.	40

Parameter	Details	Default value
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
CloudMetaCacheSize	Default meta cache size when adding Cloud LSU. Decrease this value if enough free space is not available.	500 GiB

Parameter	Details	Default value
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%

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 --verbosecloud` 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 --verbosecloud` output:

```
# crcontrol --dsstat 2 --verbosecloud

=============== 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.

After disaster recovery for cloud LSU, importing backup images is needed with following cases:

- Master has no catalog of images in MSDP storage. For example, when master is reinstalled and catalog in master is lost, it's needed to do backup images importing. Refer section "About importing backup images" in *NetBackup Administrator's Guide, Volume I* for more information.
- Master has incorrect catalog of images in MSDP storage. MSDP storage server resides on media server. When disable recovery is done by using a new media server, the new MSDP storage server resides on the new media server. At that case the catalog in master is incorrect, for the catalog still refer to old MSDP storage server which is not available. To correct the catalog in master, delete the old catalog and import backup images from the new MSDP storage server. The new media server here means a new added media server or other existing media server.
- When master 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.
- Backup images importing is not supported when the cloud LSU is based on Amazon S3 Glacier, Deep Archive, and Microsoft Azure Archive.
- Cloud LSU of Amazon S3 Glacier, Deep Archive, and Microsoft Azure Archive supports cloud disaster recovery only in Scenario 1 and Scenario 3.

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 catalog is not available in master server.

Scenario 1: The local storage is lost and images importing is not needed

Step	Task	Procedure
1	Create an empty local LSU	See Configure/Reconfigure MSDP local storage
2	Reuse Cloud LSU	See Reuse cloud LSU

Scenario 2: The local storage is lost and images importing is needed

Step	Task	Procedure
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 NetBackup Administrator's Guide, Volume I.
2	Delete old storage server-related configurations	See "Recovering from an MSDP storage server failure" on page 466. Delete the storage units that use the disk pool. Delete the disk pool. Delete the deduplication storage server. Delete the deduplication host configuration file.
3	Configure new storage server.	`/usr/openv/netbackup/bin/admincmd/nbdevconfig -creatests -storage_server "storage server" -stype PureDisk -media_server "media server" -st 9`
4	Create an empty local LSU.	See Configure/Reconfigure MSDP local storage
5	Reuse cloud LSU.	See Reuse cloud LSU
6	Create disk pool for cloud LSUs.	`/usr/openv/netbackup/bin/admincmd/nbdevconfig -createdp -stype PureDisk -dp dpname -storage_server sts_hostname -dvlist filename`
7	Import the images back.	Do two-phase import. See NetBackup Administrator's Guide, Volume I

Scenario 3: The local storage is not lost and images importing is not needed

Step	Task	Procedure
1	Reuse existing local storage path	See Configure/Reconfigure MSDP local storage
2	Restart storage server.	`/usr/openv/netbackup/bin/bp.kill_all` `/usr/openv/netbackup/bin/bp.start_all`

Scenario 4: The local storage is not lost and images importing is needed

Step	Task	Procedure
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 NetBackup Administrator's Guide, Volume I
2	Delete old storage server-related configurations.	See "Recovering from an MSDP storage server failure" on page 466. Delete the storage units that use the disk pool. Delete the disk pool. Delete the deduplication storage server. Delete the deduplication host configuration file.
3	Configure new storage server.	`/usr/openv/netbackup/bin/admincmd/nbdevconfig -creatests -storage_server "storage server" -stype PureDisk -media_server "media server" -st 9`
4	Reuse existing local storage path	See Configure/Reconfigure MSDP local storage
5	Restart storage server.	`/usr/openv/netbackup/bin/bp.kill_all` `/usr/openv/netbackup/bin/bp.start_all`
6	Create disk pool for cloud LSUs.	`/usr/openv/netbackup/bin/admincmd/nbdevconfig -createdp -stype PureDisk -dp dpname -storage_server sts_hostname -dvlist filename`
7	Import the images back.	Do two-phase import. See NetBackup Administrator's Guide, Volume I

Common disaster recovery steps

Following are the common disaster recovery steps:

- Configure/Reconfigure MSDP local storage
- Reuse cloud LSU

Configure/Reconfigure MSDP local storage

Step	Task	Procedure
1	Delete the deduplication configuration.	`/usr/openv/pdde/pdconfigure/scripts/installers/PDDE_deleteConfig.sh`
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`

Step	Task	Procedure

4	Generate the config template.	Make changes to the template file `local-lsu.txt` and delete all other entries that are not needed.
		
		Parameters without KMS encryption:
		
		`/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`
		
		Parameters with KMS encryption:
		
		`/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`
		`V7.5 "encryption" "true" int`
		`V7.5 "kmsenabled" "true" int`
		`V7.5 "kmsservertype" "0" int`
		`V7.5 "kmsservername" "KMS-SERVER-FQDN" string`
		`V7.5 "keygroupname" "key-group-name_for_this_msdp" string`
		
		Note: You can get the KMS encryption information from the cloud storage `<bucket>/subBucket/databases/kms_5Fcloud.cfg.<dsid>/0`
		
		For more information about the parameters, See "Editing an MSDP storage server configuration file" on page 227.

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 "lsuCloudUser" "your-acct-id" string`
		`V7.5 "lsuCloudPassword" "your-secret" string`
		`V7.5 "lsuCloudAlias" "<storageserver_lsuname>" string`
		`V7.5 "lsuCloudBucketName" "bucket-name" string`
		`V7.5 "lsuCloudBucketSubName" "lsuname" string`
		Configuration template example 2 with encryption enabled:
		`V7.5 "operation" "reuse-lsu-cloud" string`
		`V7.5 "lsuCloudUser" "your-acct-id" string`
		`V7.5 "lsuCloudPassword" "your-secret" 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`

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 alias do 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>`
		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`

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 execute 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 disd value is “0”, local LSU is processed.

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 "lsuCloudUser" "your-acct-id" string
V7.5 "lsuCloudPassword" "your-secret" 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 "lsuCloudUser" "your-acct-id" string
V7.5 "lsuCloudPassword" "your-secret" 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 281.

About Image Sharing using MSDP cloud

Use image sharing to share the images from your on-premises NetBackup server to the NetBackup server running in AWS or Azure. The NetBackup server that is running in the cloud and 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.

The following table describes the image sharing feature workflow.

Table 7-1 Image sharing workflow

Task	Description
Prepare 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
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 cloud recovery server. Perform the following step to configure the image sharing: ■ Configure Image sharing using MSDP cloud by NetBackup Web UI ■ Configure Image sharing using MSDP cloud with the ims_system_config.py script

Table 7-1 Image sharing workflow *(continued)*

Task	Description
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

- In a situation where MSDP cloud backed up the deduplicated data to cloud, but the NetBackup catalog was available only on the on-premises NetBackup server. There, the data cannot be restored from the cloud without the on-premises NetBackup server.
  Image sharing in 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 an all-in-one NetBackup in the cloud on demand called the cloud recovery server, and recover the backup images from 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 in cloud” on page 332.

Things to consider before you use image sharing

- Before you install NetBackup, create an instance based on RHEL 7.3 or later in cloud. You can also set up a computer based on 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 host name to the server's FQDN.
  In Azure virtual machine, you must change the internal hostname, which is created automatically for you and cannot get internal hostname from 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.
- NetBackup should be an all-in-one setup.
  Refer to the *NetBackup Installation Guide* for more information.

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 cloud recovery server for image sharing* topic in the *NetBackup Web UI Administrator's Guide*.

Configure Image sharing using MSDP cloud with the ims_system_config.py script

After installing NetBackup, you can run the `ims_system_config.py` script to configure image sharing.

The path to access the command is: `/usr/openv/pdde/pdag/scripts/`.

Amazon Web Service cloud provider:

```
ims_system_config.py -t PureDisk -k <AWS_access_key> -s
<AWS_secret_access_key> -b <name_S3_bucket> -bs <bucket_sub_name>
[-r <bucket_region>] [-p <mount_point>]
```

If you have configured IAM role in the EC2 instance, use the following command:

```
ims_system_config.py  -t PureDisk -k dummy -s dummy <bucket_name>
-bs <bucket_sub_name> [-r <bucket_region>] [-p <mount_point>]
```

Microsoft Azure cloud provider:

```
ims_system_config.py -cp 2 -k <key_id> -s <secret_key> -b
<container_name> -bs <bucket_sub_name> [-p <_mount_point_>]
```

Other S3 compatible cloud provider (For example, Hitachi HCP):

If Cloud Instance has been existed in NetBackup, use the following command:

```
ims_system_config.py -cp 3 -t PureDisk -k <key_id> -s <secret_key>
-b <bucket_name> -bs <bucket_sub_name> -c <Cloud_instance_name> [-p
 <mount_point>]
```

Or use the following command:

```
ims_system_config.py -cp 3 -t PureDisk -k <key_id> -s <secret_key>
-b <bucket_name> -pt <cloud_provider_type> -sh <s3_hostname> -sp
<s3_http_port> -sps <s3_https_port> -ssl <ssl_usage> [-p
<mount_point>]
```

Example for HCP provider:

```
ims_system_config.py -cp 3 -t PureDisk -k xxx -s xxx -b emma -bs
subtest -pt hitachicp  -sh yyy.veritas.com -sp 80 -sps 443 -ssl 0
```

Description: (Specify the following options to use HCP cloud)

`-cp 3`: Specify third-party S3 cloud provider that is used.

`-pt hitachicp`: Specify cloud provider type as **hitachicp** (HCP LAN)

`-t PureDisk_hitachicp_rawd`: Specify storage server type as PureDisk_hitachicp_rawd

`-sh <s3_hostname>`: Specify HCP storage server host name

`-sp <s3_http_port>`: Specify HCP storage server HTTP port (Default is 80)

`-sps <s3_https_port>`: Specify HCP storage server HTTP port (Default is 443)

`-ssl <ssl_usage>`: Specify whether to use SSL. (0- Disable SSL. 1- Enable SSL. Default is 1.) If SSL is disabled, it uses `<s3_http_port>` to make connection to <s3_hostname>. Otherwise, it uses `<s3_https_port>`.

Using image sharing by NetBackup Web UI

You can access NetBackup Web UI to use image sharing. For more information, refer to the *Using image sharing from the NetBackup Web UI* 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 7-2 Steps for image sharing and the command options

Step	Command

Log on to NetBackup	`nbimageshare --login <username> <password>`
	
	`nbimageshare --login -interact`

List all the backup images that are in the cloud	`nbimageshare --listimage` **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>`
	
	Import multiple images:
	
	`nbimageshare --batchimport <image_list_file_path>`
	
	Note: The format of the `image_list_file_path` is same as the output of "list images".
	
	The multiple images number must be equal to or less than 64.
	
	You can import an already imported image. This action does not affect the NetBackup image catalog.

Table 7-2 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 <client> <policy> <backupID>`
	■ Only VM images are supported.
	■ 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 key group name is in `contentrouter.cfg` under `[KMSOptions]`

   (Example `KMSKeyGroupName=amazon.com:test1`)

2. NetBackup master 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. Config 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 or with `ims_system_config.py`

On-premises KMS key changes:

In 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 on-premises KMS server and import that key file on the cloud recovery server.

1. On-premises NetBackup master 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 external KMS server, then make sure that the same KMS server is configured on the cloud recovery server before running `ims_system_config.py`. 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 in the cloud on demand and don't 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). If KMS encryption is enabled in on-premise side after image sharing server is configured, the encrypted image cannot be import by this image sharing server.
- Cloud LSU requires free disk space. When you configure image sharing server using the `ims_system_config.py` script, ensure that you have enough disk space in the default mount point or storage, or you can use `-p` parameter of `ims_system_config.py` to specify a different mount point to meet the requirement of free disk spaces.
- 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.
- 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 instance 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 lists for the latest information on the supported policy types.
- After the image sharing is configured, the storage server is in a read-only mode.
- 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.
- 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 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.
- If you want to convert a VM image again, you must delete the VHD from 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 VM based on VHD. Image sharing does not add additional limitation about VM conversion, but Azure has the following prerequisites on source VMs:

- Source virtual machine OS Type
  Following guest operating systems in source virtual machine are supported:
  - Windows 10 Series
  - Windows 2012 R2 Series
  - Windows 2016 Series
  - Windows 2019 Series
  - RHEL 7.6, 7.7
  - Ubuntu 18.04
  - SUSE 12SP4

  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 boot 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.

- Boot and partition type of source virtual machine

  The source VM must boot using BIOS. The OS volume must use MBR partitioning rather than GPT.

- 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 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 system. 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 new VM created by converted VHD.

Linux: The name of 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 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.

Converting the VM image to VHD in Azure

Windows 2016

To convert the Windows 2016 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 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.
   - Login the converted VM through RDP.

RHEL7.6

Pre-requisites:

- Source VM OS volume must use MBR partitioning rather than GPT.

- It is recommended that persistent naming is used and the filesystem label or UUID for Azure Linux VMs is used.
  Most distributions provide the `fstab nofail` or `nobootwait` parameters. These parameters enable a system to boot when the disk fails to mount at startup.
- Enure that OS is installed on the first disk of source VM and do not configure a swap partition on the operating system disk. see Information for Non-endorsed Distributions.
- It is recommended that the network interface in 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 `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 enabled on boot. After this change, reboot VM to check if eth0 works.

   In the network interface configuration file, configure: `ONBOOT=yes`.

   The example to change the network interface to eth0:

   ```
   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 image sharing server and configure image sharing feature with azure account.
6. Import the backup image and perform the conversion.
7. 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.
   - Login the converted VM through RDP.

SUSE 12 SP4

Pre-requisites:

- Source VM OS volume must use MBR partitioning rather than GPT.
- It is recommended that persistent naming is used and the filesystem label or UUID for Azure Linux VMs is used.
  Most distributions provide the `fstab nofail` or `nobootwait` parameters. These parameters enable a system to boot when the disk fails to mount at startup.
- Enure that OS is installed on the first disk of source VM and do not configure a swap partition on the operating system disk. see Information for Non-endorsed Distributions.

- It is recommended that the network interface in 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. Make sure 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 image sharing server and configure image sharing feature with azure account.
5. Import the backup image and perform the conversion.
6. 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.
   - Login the converted VM through RDP.

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 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 Linux operating system 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 308.
- Amazon S3 compatible storages
  See “About immutable object support for AWS S3 compatible platforms” on page 315.
- Microsoft Azure immutable storage
  See “About immutable storage support for Azure blob storage ” on page 325.

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 329.

About MSDP cloud admin tool

Use the MSDP cloud admin tool **msdpcldutil** to manage the cloud immutable volume. This tool is located at `/usr/openv/pdde/pdcr/bin/msdpcldutil`. Cloud administrator, who has the required permissions can run this tool. You can run this tool from NetBackup primary server, NetBackup media server, or MSDP storage server on Red Hat Linux operating system.

You can use this tool to perform the following tasks for cloud immutable storage:

- Create the cloud immutable volume.
- List the volumes.

- Update the cloud immutable volume min and max retention period.
- Update the cloud immutable volume live duration.
- List cloud immutable storage cloud providers.
- Switch the retention mode from governance mode to compliance mode.
  The compliance mode is stricter than governance mode. Some users also use governance mode to test retention period settings before creating a compliance mode retention period.

See "Managing AWS S3 immutable storage using msdpcldutil tool" on page 311.

See "Managing HCP for Cloud Scale using msdpcldutil tool" on page 318.

See "Managing Cloudian HyperStore using msdpcldutil tool" on page 318.

See "Managing Seagate Lyve Cloud using msdpcldutil tool" on page 319.

See "Managing Veritas Access Cloud using msdpcldutil tool " on page 321.

See "Managing an Azure cloud immutable volume using msdpcldutil tool" on page 327.

See "Troubleshooting the error when the bucket is created without msdpcldutil " on page 330.

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 313.

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.
- Governance mode (also known as 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 governance mode to test the retention period behavior before you use compliance mode.

You can use MSDP cloud admin tool to manage cloud immutable volume.

See "Managing AWS S3 immutable storage using msdpcldutil tool" on page 311.

Cloud immutable volume (Cloud LSU) is a cloud volume with the following differences than normal cloud volumes:

- The bucket is Object Lock enabled. It is created with the tool `msdpcldutil`.
- The bucket policy is attached to the bucket to protect metadata objects of cloud immutable volume.
- 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. This range can be defined and changed with `msdpcldutil`.
- The cloud volume has a live period which defines its lifetime. It provides a safety net so that the retention period of all data in it is restricted in the lifetime of cloud volume. When this live period expires, the volume will be down. You can use `msdpcldutil` to extend the live period when the volume does not expire or resurrect the volume when it expires.
- You can set the storage class while creating immutable volume. Use the option `--storageclass` with your storage class, such as `STANDARD_IA` or `GLACIER_IR`. The default value is STANDARD.

Creating a cloud immutable storage unit

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.

To create a cloud immutable storage unit

1. Use `msdpcldutil` command to create the cloud immutable volume. Note down the volume name, it will be used in step 4.

   See "Managing AWS S3 immutable storage using msdpcldutil tool" on page 311.

   Make sure that Amazon cloud administrators have the required permissions. See "AWS user permissions to create the cloud immutable volume" on page 313.

2. On the NetBackup Web UI, navigate to **Storage > Disk pools**, and click **Add**.

3 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**.

4 From the **Volume** drop-down list select a volume or add a new volume. Provide the name created in step 1 by msdpcldutil.

On the **Cloud storage provider** window, select **Amazon** 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**.

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 you provide the bucket name, ensure this bucket is created by msdpcldutil so that it is Object Lock enabled.

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**.

5 In **Replication**, click **Next**.

6 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.

7 In the **Storage unit** tab, click **Add**.

8 Select **Media Server Deduplication Pool (MSDP)** and click **Start**.

9 In **Basic properties**, enter the **Name** of the MSDP storage unit and click **Next**.

10 Select the disk pool that was created and select the **Enable WORM/Lock until expiration** box, and click **Next**.

11 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.

12 Review the setup of the storage unit and then click **Save**.

Managing AWS S3 immutable storage using msdpcldutil tool

MSDP cloud admin tool `/usr/openv/pdde/pdcr/bin/msdpcldutil` is used to manage immutable.

Before using this tool, set the following environment variables:

```
export MSDPC_ACCESS_KEY=xxxx
export MSDPC_SECRET_KEY=yyyyyyyyyyyy
export MSDPC_REGION=us-east-1
export MSDPC_PROVIDER=amazon
```

For Amazon S3, `MSDPC_ACCESS_KEY` is the AWS access key associated with an IAM user. `MSDPC_SECRET_KEY` is the secret key associated with the access key. `MSDPC_REGION` is the AWS region where the bucket will be created or accessed.

If NetBackup media server is deployed on Amazon EC2 instance, you can use IAM role-based authentication after the IAM role configuration. You must have the cloud administrator permissions to create the IAM role policy and attach the IAM policy to the role.

See "AWS user permissions to create the cloud immutable volume" on page 313.

Perform the following tasks to create the immutable storage and configure it:

- Create a cloud immutable volume.
  ```
  #/usr/openv/pdde/pdcr/bin/msdpcldutil create -b bucketname -v
  volumename --mode GOVERNANCE --min 1D --max 30D --live 2021-12-31
  ```
- List the cloud volumes.
  ```
  #/usr/openv/pdde/pdcr/bin/msdpcldutil list
  ```
- Update the cloud immutable volume mode.
  ```
  #/usr/openv/pdde/pdcr/bin/msdpcldutil update mode -b bucketname
  -v volumename --mode COMPLIANCE --live 2021-12-31 --inherit enable
  ```
  - `--inherit disable` If your Governance mode data is testing and does not need to be protected, you must use this option.

- `--inherit enable` If you want to protect the Governance mode data, you must use this option.

The volume retention mode can switch from governance to compliance. It cannot switch from compliance to governance. After governance mode is switched to the compliance mode, the new backup image retention mode is compliance. When the mode is switched from governance to compliance mode, due to the nature of deduplication, the images in compliance mode may share some data in the previous images in governance data. Users then have a choice to lock this shared data in either existing governance mode or in compliance mode.

- Update the cloud immutable volume min and max retention period.
  - `#/usr/openv/pdde/pdcr/bin/msdpcldutil update range -b bucketname -v volumename --min 1D --max 90D`
  - `# /usr/openv/netbackup/bin/admincmd/nbdevconfig -updatedv -stype PureDisk -dp disk_pool_name -dv volumename`

  The minimum and maximum values are defined by the min and max options. Both values must be between 1 day and 30 years. The maximum value must be less than the volume live duration.

- Update the cloud immutable volume live duration.

  `#/usr/openv/pdde/pdcr/bin/msdpcldutil update live -b bucketname -v volumename -l 2022-01-31`

  The volume has live period property, which is a timestamp. The backup image retention time must be less than this timestamp. If the live period expires, the volume stops and the backup job fails with the following error message in job details:

  `Critical bptm (pid=xxxxx) Failed to set WORM immutable and indelible lock for image: clientname_1620671199_C1_IM with status: 2060404 Attempt to WORM lock data past the configured MSDP Cloud lifetime`

  Cloud administrator can bring the volume back to the running state by extending the live period. You can try the job again.

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:GetBucketPolicyStatus",
"s3:GetObjectRetention",
"s3:DeleteObjectVersion",
"s3:ListBucketVersions",
"s3:CreateBucket",
"s3:ListBucket",
"s3:GetBucketVersioning",
"s3:BypassGovernanceRetention",
"s3:GetBucketPolicy",
"s3:GetBucketObjectLockConfiguration",
"s3:PutObject",
```

```
"s3:GetObject",
"s3:ListAllMyBuckets",
"s3:PutObjectRetention",
"s3:PutBucketPolicy",
"s3:PutBucketObjectLockConfiguration",
"s3:DeleteObject",
"s3:GetBucketLocation",
"s3:DeleteBucket",
"s3:DeleteBucketPolicy",
"s3:PutBucketVersioning",
"s3:GetObjectVersion"
```

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:GetObjectRetention",
"s3:DeleteObjectVersion",
"s3:ListBucketVersions",
"s3:ListBucket",
"s3:GetBucketVersioning",
"s3:GetBucketObjectLockConfiguration",
"s3:PutObject",
"s3:GetObject",
"s3:ListAllMyBuckets",
"s3:PutObjectRetention",
"s3:DeleteObject",
"s3:GetBucketLocation",
"s3:GetObjectVersion",
"s3:BypassGovernanceRetention",
```

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. Bucket policy is created and updated automatically when cloud immutable volume is created.

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 “Troubleshooting the error when the bucket is created without msdpcldutil ” on page 330.

See “AWS user permissions to create the cloud immutable volume” on page 313.

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.

- 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.

From NetBackup 10.0.1 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 – LAN (version 8.5)
  - Cloud admin role and backup admin role are combined into a single role.
  - Only compliance mode is supported.
- EMC-ECS (vesion 3.6.2)
  Cloud admin role and backup admin role are combined into a single role.

Creating a cloud immutable storage unit for the S3 compatible platforms

Use the NetBackup Web UI to create a cloud immutable storage unit. The following steps describe the method to create a cloud immutable storage unit for HCP for Cloud Scale, Cloudian HyperStore, Seagate Lyve Cloud, and Veritas Access Cloud.

To create a cloud immutable storage unit:

1. Use `msdpcldutil` command to create the cloud immutable volume. Note down the volume name, it will be used in step 4.
2. On the NetBackup Web UI, navigate to **Storage > Disk pools**, and click **Add**.
3. 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**.

4 From the **Volume** drop-down list select a volume or add a new volume. Provide the name that is created in step 1 by **msdpcldutil**.

On the **Cloud storage provider** window, select **Hitachi Content Platform for Cloud Scale - LAN**, **Hitachi Content Platform for Cloud Scale - WAN**, **Seagate Lyve Cloud**, or **Veritas Access Cloud** 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**.

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 you provide the bucket name, ensure this bucket is created by **msdpcldutil** so that it is Object Lock enabled.

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**.

5 In **Replication**, click **Next**.

6 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.

7 In the **Storage unit** tab, click **Add**.

8 Select **Media Server Deduplication Pool (MSDP)** and click **Start**.

9 In **Basic properties**, enter the **Name** of the MSDP storage unit and click **Next**.

10 Select the disk pool that was created and select the **Enable WORM/Lock until expiration** box, and click **Next**.

11 In **Media server**, use the default selection of **Allow NetBackup to automatically select**, and click **Next**.

12 Review the setup of the storage unit and then click **Save**.

Managing HCP for Cloud Scale using msdpcldutil tool

MSDP cloud admin tool `/usr/openv/pdde/pdcr/bin/msdpcldutil` is used to manage cloud immutable volume.

Set the following environment variables:

```
export MSDPC_ACCESS_KEY=xxxx
export MSDPC_SECRET_KEY=yyyy
export MSDPC_REGION=us-west-2
export MSDPC_PROVIDER= hitachi-csw
export MSDPC_ENDPOINT=hcpcloudscale.hostname
```

HCP for Cloud Scale has two types of providers:

- Hitachi-csw (Hitachi Cloud Scale, WAN)
- Hitachi-csl (Hitachi Cloud Scale, LAN)

To create the immutable storage and configure it:

1. Create a cloud immutable volume.

   ```
   #/usr/openv/pdde/pdcr/bin/msdpcldutil create -b bucketname -v
   volumename --mode COMPLIANCE --min 1D --max 30D --live 2021-12-31
   ```

2. List the cloud volumes.

   ```
   #/usr/openv/pdde/pdcr/bin/msdpcldutil list
   ```

3. Update the cloud immutable volume min and max retention period.

   - `#/usr/openv/pdde/pdcr/bin/msdpcldutil update range -b bucketname -v volumename --min 1D --max 90D`
   - `# /usr/openv/netbackup/bin/admincmd/nbdevconfig -updatedv -stype PureDisk -dp disk_pool_name -dv volumename`

4. Update the cloud immutable volume live duration.

   ```
   #/usr/openv/pdde/pdcr/bin/msdpcldutil update live -b bucketname
   -v volumename -l 2022-01-31
   ```

5. List cloud immutable storage cloud providers.

   ```
   #/usr/openv/pdde/pdcr/bin/msdpcldutil platform list
   ```

Managing Cloudian HyperStore using msdpcldutil tool

MSDP cloud admin tool `/usr/openv/pdde/pdcr/bin/msdpcldutil` is used to manage cloud immutable volume.

Set the following environment variables:

```
export MSDPC_ACCESS_KEY=xxxx
export MSDPC_SECRET_KEY=yyyy
export MSDPC_REGION=us-east-1
export MSDPC_PROVIDER=cloudian
export MSDPC_ENDPOINT=cloudian.hyperstore.hostname
```

Cloudian HyperStor has one provider - cloudian (Cloudian HyperStore)

To create the immutable storage and configure it:

1. Create a cloud immutable volume.
   - • `#/usr/openv/pdde/pdcr/bin/msdpcldutil create -b bucketname -v volumename --mode COMPLIANCE --min 1D --max 30D --live 2022-12-31`
   - • `#/usr/openv/pdde/pdcr/bin/msdpcldutil create -b bucketname -v volumename --mode GOVERNANCE --min 1D --max 30D --live 2022-12-31`
2. List the cloud volumes.

   `#/usr/openv/pdde/pdcr/bin/msdpcldutil list`
3. Update the cloud immutable volume min and max retention period.
   - `#/usr/openv/pdde/pdcr/bin/msdpcldutil update range -b bucketname -v volumename --min 1D --max 90D`
   - `# /usr/openv/netbackup/bin/admincmd/nbdevconfig -updatedv -stype PureDisk -dp disk_pool_name -dv volumename`
4. Update the cloud immutable volume live duration.

   `#/usr/openv/pdde/pdcr/bin/msdpcldutil update live -b bucketname -v volumename -l 2022`
5. Update the cloud immutable volume mode.

   `##/usr/openv/pdde/pdcr/bin/msdpcldutil update mode -b bucketname -v volumename --mode COMPLIANCE --live 2022-12-31 --inherit enable`
6. List cloud immutable storage cloud providers.

   `#/usr/openv/pdde/pdcr/bin/msdpcldutil platform list`

Managing Seagate Lyve Cloud using msdpcldutil tool

MSDP cloud admin tool `/usr/openv/pdde/pdcr/bin/msdpcldutil` is used to manage cloud immutable volume.

Before using this tool, set the following environment variables:

```
export MSDPC_ACCESS_KEY=xxxx
export MSDPC_SECRET_KEY=yyyy
export MSDPC_REGION=us-west-1
export MSDPC_PROVIDER=seagate
export MSDPC_ENDPOINT=seagate-lyve-cloud.hostname
```

Seagate Lyve Cloud has one provider – Seagate (Seagate Lyve Cloud)

To create the immutable storage and configure it

1. Create a cloud immutable volume.
   - • #/usr/openv/pdde/pdcr/bin/msdpcldutil create -b bucketname -v volumename --mode COMPLIANCE --min 1D --max 30D --live 2022-12-31
   - • #/usr/openv/pdde/pdcr/bin/msdpcldutil create -b bucketname -v volumename --mode GOVERNANCE --min 1D --max 30D --live 2022-12-31
2. List the cloud volumes.

   ```
   #/usr/openv/pdde/pdcr/bin/msdpcldutil list
   ```
3. Update the cloud immutable volume min and max retention period.
   - #/usr/openv/pdde/pdcr/bin/msdpcldutil update range -b bucketname -v volumename --min 1D --max 90D
   - # /usr/openv/netbackup/bin/admincmd/nbdevconfig -updatedv –stype PureDisk -dp disk_pool_name -dv volumename
4. Update the cloud immutable volume live duration.

   ```
   #/usr/openv/pdde/pdcr/bin/msdpcldutil update live -b bucketname -v volumename -l 2022-01-31
   ```
5. Update the cloud immutable volume mode.

   ```
   ##/usr/openv/pdde/pdcr/bin/msdpcldutil update mode -b bucketname -v volumename --mode COMPLIANCE --live 2022-12-31 --inherit enable
   ```
6. List cloud immutable storage cloud providers.

   ```
   #/usr/openv/pdde/pdcr/bin/msdpcldutil platform list
   ```

Managing Veritas Access Cloud using msdpcldutil tool

MSDP cloud admin tool `/usr/openv/pdde/pdcr/bin/msdpcldutil` is used to manage cloud immutable volume.

Before using this tool, set the following environment variables:

```
export MSDPC_ACCESS_KEY=xxxx
export MSDPC_SECRET_KEY=yyyy
export MSDPC_REGION=us-east-1
export MSDPC_PROVIDER=vtas-access
export MSDPC_ENDPOINT=veritas_access.hostname
```

Veritas Access Cloud has one provider – Veritas

To create the immutable storage and configure it

1. Create a cloud immutable volume.
   - `#/usr/openv/pdde/pdcr/bin/msdpcldutil create -b bucketname -v volumename --mode COMPLIANCE --min 1D --max 30D --live 2022-12-31`
   - `#/usr/openv/pdde/pdcr/bin/msdpcldutil create -b bucketname -v volumename --mode GOVERNANCE --min 1D --max 30D --live 2022-12-31`
2. List the cloud volumes.

   `#/usr/openv/pdde/pdcr/bin/msdpcldutil list`
3. Update the cloud immutable volume min and max retention period.
   - `#/usr/openv/pdde/pdcr/bin/msdpcldutil update range -b bucketname -v volumename --min 1D --max 90D`
   - `# /usr/openv/netbackup/bin/admincmd/nbdevconfig -updatedv –stype PureDisk -dp disk_pool_name -dv volumename`
4. Update the cloud immutable volume live duration.

   `#/usr/openv/pdde/pdcr/bin/msdpcldutil update live -b bucketname -v volumename -l 2022-01-31`

5 Update the cloud immutable volume mode.

```
##/usr/openv/pdde/pdcr/bin/msdpcldutil update mode -b bucketname
-v volumename --mode COMPLIANCE --live 2022-12-31 --inherit enable
```

6 List cloud immutable storage cloud providers.

```
#/usr/openv/pdde/pdcr/bin/msdpcldutil platform list
```

Managing Wasabi cloud storage using msdpcldutil tool

MSDP cloud admin tool `/usr/openv/pdde/pdcr/bin/msdpcldutil` is used to manage cloud immutable volume.

Before using this tool, set the following environment variables:

```
export MSDPC_ACCESS_KEY=wasabi-access-key
export MSDPC_SECRET_KEY=wasabi-secret-key
export MSDPC_REGION=us-west-2
export MSDPC_PROVIDER=wasabi
export MSDPC_ENDPOINT=wasabi.hostname
```

To create the immutable storage and configure it

1 Create a cloud immutable volume.

```
#/usr/openv/pdde/pdcr/bin/msdpcldutil create -b bucketname -v
volumename --mode COMPLIANCE --min 1D --max 30D --live 2021-12-31
```

2 List the cloud volumes.

```
#/usr/openv/pdde/pdcr/bin/msdpcldutil list
```

3 Update the cloud immutable volume minimum and maximum retention period.

- `#/usr/openv/pdde/pdcr/bin/msdpcldutil update range -b bucketname -v volumename --min 1D --max 90D`
- `# /usr/openv/netbackup/bin/admincmd/nbdevconfig -updatedv -stype PureDisk -dp disk_pool_name -dv volumename`

4 Update the cloud immutable storage volume live duration.

```
#/usr/openv/pdde/pdcr/bin/msdpcldutil update live -b bucketname
-v volumename -l 2022-01-31
```

5 List cloud immutable storage cloud providers.

```
#/usr/openv/pdde/pdcr/bin/msdpcldutil platform list
```

Managing Scality RING – LAN using msdpcldutil tool

MSDP cloud admin tool `/usr/openv/pdde/pdcr/bin/msdpcldutil` is used to manage cloud immutable volume.

Before using this tool, set the following environment variables:

```
export MSDPC_ACCESS_KEY=scalitylan-access-key
export MSDPC_SECRET_KEY=scalitylan-secret-key
export MSDPC_REGION=us-east-1
export MSDPC_PROVIDER=scalitylan
export MSDPC_ENDPOINT=scalitylan.hostname
```

To create the immutable storage and configure it

1. Create a cloud immutable volume.
   - `#/usr/openv/pdde/pdcr/bin/msdpcldutil create -b bucketname -v volumename --mode COMPLIANCE --min 1D --max 30D --live 2021-12-31`
   - `#/usr/openv/pdde/pdcr/bin/msdpcldutil create -b bucketname -v volumename --mode GOVERNANCE --min 1D --max 30D --live 2022-12-31`
2. List the cloud volumes.

   `#/usr/openv/pdde/pdcr/bin/msdpcldutil list`
3. Update the cloud immutable volume minimum and maximum retention period.
   - `#/usr/openv/pdde/pdcr/bin/msdpcldutil update range -b bucketname -v volumename --min 1D --max 90D`
   - `# /usr/openv/netbackup/bin/admincmd/nbdevconfig -updatedv –stype PureDisk -dp disk_pool_name -dv volumename`
4. Update the cloud immutable volume live duration.

   `#/usr/openv/pdde/pdcr/bin/msdpcldutil update live -b bucketname -v volumename -l 2022-01-31`
5. Update the cloud immutable storage volume mode.

   `##/usr/openv/pdde/pdcr/bin/msdpcldutil update mode -b bucketname -v volumename --mode COMPLIANCE --live 2022-12-31 --inherit enable`
6. List cloud immutable storage cloud providers.

   `#/usr/openv/pdde/pdcr/bin/msdpcldutil platform list`

Managing EMC-ECS using msdpcldutil tool

MSDP cloud admin tool `/usr/openv/pdde/pdcr/bin/msdpcldutil` is used to manage cloud immutable volume.

Before using this tool, set the following environment variables:

```
export MSDPC_ACCESS_KEY=emcecs-access-key
export MSDPC_SECRET_KEY=emcecs-secret-key
export MSDPC_REGION=us-east-1
export MSDPC_PROVIDER=EMC-ECS
export MSDPC_ENDPOINT=emcecs.hostname
```

To create the immutable storage and configure it

1. Create a cloud immutable volume.
   - `#/usr/openv/pdde/pdcr/bin/msdpcldutil create -b bucketname -v volumename --mode COMPLIANCE --min 1D --max 30D --live 2021-12-31`
   - `#/usr/openv/pdde/pdcr/bin/msdpcldutil create -b bucketname -v volumename --mode GOVERNANCE --min 1D --max 30D --live 2022-12-31`
2. List the cloud volumes.

   `#/usr/openv/pdde/pdcr/bin/msdpcldutil list`
3. Update the cloud immutable volume minimum and maximum retention period.
   - `#/usr/openv/pdde/pdcr/bin/msdpcldutil update range -b bucketname -v volumename --min 1D --max 90D`
   - `# /usr/openv/netbackup/bin/admincmd/nbdevconfig -updatedv -stype PureDisk -dp disk_pool_name -dv volumename`
4. Update the cloud immutable volume live duration.

   `#/usr/openv/pdde/pdcr/bin/msdpcldutil update live -b bucketname -v volumename -l 2022-01-31`
5. Update the cloud immutable storage volume mode.

   `##/usr/openv/pdde/pdcr/bin/msdpcldutil update mode -b bucketname -v volumename --mode COMPLIANCE --live 2022-12-31 --inherit enable`
6. List cloud immutable storage cloud providers.

   `#/usr/openv/pdde/pdcr/bin/msdpcldutil platform list`

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 locked policy for the defined retention period. Once you set a retention period for the data storage, you can extend, shorten, or delete it.

Creating an Azure cloud immutable storage unit using the Web UI

Ensure that the MSDP storage server is created before performing the following steps.

To create an Azure cloud immutable storage unit

1. Use `msdpcldutil` command to create the cloud immutable volume. Note down the volume name, it will be used in step 4.
2. On the NetBackup Web UI, navigate to **Storage > Disk pools**, and click **Add**.
3. 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**.

4 From the **Volume** drop-down list select a volume or add a new volume. Provide the name that is created in step 1 by **msdpcldutil**.

On the **Cloud storage provider** window, select **Microsoft Azure** 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**.

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 you provide the bucket name, ensure this bucket is created by **msdpcldutil** so that it is Object Lock enabled.

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**.

5 In **Replication**, click **Next**.

6 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.

7 In the **Storage unit** tab, click **Add**.

8 Select **Media Server Deduplication Pool (MSDP)** and click **Start**.

9 In **Basic properties**, enter the **Name** of the MSDP storage unit and click **Next**.

10 Select the disk pool that was created and select the **Enable WORM/Lock until expiration** box, and click **Next**.

11 In **Media server**, use the default selection of **Allow NetBackup to automatically select**, and click **Next**.

12 Review the setup of the storage unit and then click **Save**.

Managing an Azure cloud immutable volume using msdpcldutil tool

MSDP cloud admin tool `/usr/openv/pdde/pdcr/bin/msdpcldutil` is used to manage cloud immutable volume.

You can create an Azure cloud immutable volume in the following scenarios:

- Azure storage account has enabled version-level immutability support.
- The container is created through Azure portal and has enabled version-level immutability support.
- You use Azure service principal.

For information about immutability policies configurations in Azure, see Configure immutability policies for blob versions

To create a cloud volume when version-level immutability support is enabled:

1. Set the following environment variables:

```
# export MSDPC_REGION=<your region>
# export MSDPC_PROVIDER=azure
# export MSDPC_ACCESS_KEY=<your storage account>
# export MSDPC_SECRET_KEY=<your access key>
# export MSDPC_ENDPOINT=https://<your storage acount>.blob.core.windows.n
```

2. Create a cloud immutable volume.

```
# msdpcldutil create -b bucketname -v volumename --mode GOVERNANCE
--min 1D --max 30D --live 2022-12-31
```

   GOVERNANCE is unlocked policy and COMPLIANCE is locked policy in Azure.

3. List the cloud volumes.

```
#/usr/openv/pdde/pdcr/bin/msdpcldutil list
```

4. Update the cloud immutable volume min and max retention period.

   - `#/usr/openv/pdde/pdcr/bin/msdpcldutil update range -b bucketname -v volumename --min 1D --max 90D`
   - `# /usr/openv/netbackup/bin/admincmd/nbdevconfig -updatedv -stype PureDisk -dp disk_pool_name -dv volumename`

5 Update the cloud immutable volume live duration.

```
#/usr/openv/pdde/pdcr/bin/msdpcldutil update live -b bucketname
-v volumename -l 2023-01-31
```

6 List cloud immutable storage cloud providers.

```
#/usr/openv/pdde/pdcr/bin/msdpcldutil platform list
```

To create a cloud volume when the container is created through Azure portal and has enabled version-level immutability support:

1 Set the following environment variables:

```
# export MSDPC_REGION=<your region>
# export MSDPC_PROVIDER=azure
# export MSDPC_ACCESS_KEY=<your storage account>
# export MSDPC_SECRET_KEY=<your access key>
# export MSDPC_ENDPOINT=https://<your storage acount>.blob.core.windows.n
```

2 Create a cloud immutable volume.

```
# msdpcldutil create -b bucketname -v volumename --mode GOVERNANCE
--min 1D --max 30D --live 2022-12-31
```

GOVERNANCE is unlocked policy and COMPLIANCE is locked policy in Azure.

3 List the cloud volumes.

```
#/usr/openv/pdde/pdcr/bin/msdpcldutil list
```

4 Update the cloud immutable volume min and max retention period.

- `#/usr/openv/pdde/pdcr/bin/msdpcldutil update range -b bucketname -v volumename --min 1D --max 90D`
- `# /usr/openv/netbackup/bin/admincmd/nbdevconfig -updatedv –stype PureDisk -dp disk_pool_name -dv volumename`

5 Update the cloud immutable volume live duration.

```
#/usr/openv/pdde/pdcr/bin/msdpcldutil update live -b bucketname
-v volumename -l 2023-01-31
```

6 List cloud immutable storage cloud providers.

```
#/usr/openv/pdde/pdcr/bin/msdpcldutil platform list
```

To create a cloud volume using Azure service principal:

1. Set the following environment variables:

```
# export MSDPC_REGION=<your region>
# export MSDPC_PROVIDER=azure
# export MSDPC_ACCESS_KEY=<your storage account>
# export MSDPC_SECRET_KEY=<your access key>
# export MSDPC_ENDPOINT=https://<your storage acount>.blob.core.windows.n
# export MSDPC_SUBSCRIPTION_ID=<your subscription id >
# export MSDPC_RESOURCE_GROUP=<resource group storage acct is in>
# export AZURE_TENANT_ID=<azure tenant id>
# export AZURE_CLIENT_ID=<azure client id>
# export AZURE_CLIENT_SECRET=<azure client secret>
```

2. Create a cloud immutable volume.

```
# msdpcldutil create -b bucketname -v volumename --mode GOVERNANCE
--min 1D --max 30D --live 2022-12-31
```

   GOVERNANCE is unlocked policy and COMPLIANCE is locked policy in Azure.

3. List the cloud volumes.

```
#/usr/openv/pdde/pdcr/bin/msdpcldutil list
```

4. Update the cloud immutable volume min and max retention period.

   - `#/usr/openv/pdde/pdcr/bin/msdpcldutil update range -b bucketname -v volumename --min 1D --max 90D`
   - `# /usr/openv/netbackup/bin/admincmd/nbdevconfig -updatedv -stype PureDisk -dp disk_pool_name -dv volumename`

5. Update the cloud immutable volume live duration.

```
#/usr/openv/pdde/pdcr/bin/msdpcldutil update live -b bucketname
-v volumename -l 2023-01-31
```

6. List cloud immutable storage cloud providers.

```
#/usr/openv/pdde/pdcr/bin/msdpcldutil platform 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 307.

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 308.
- Amazon S3 compatible storage
  See “About immutable object support for AWS S3 compatible platforms” on page 315.
- Azure immutable storage
  See “About immutable storage support for Azure blob storage ” on page 325.

To use the cloud immutable storage, the following two steps are performed:

1. Use `msdpcldutil` command tool create the cloud immutable volume.
2. Use NetBackup WebUI to deploy the storage unit.

When using **msdpcldutil** command tool to update the cloud WORM volume configuration, it is recommended that you run it in the MSDP storage server engine. Otherwise, MSDP takes time to fetch the updated configurations from cloud. The maximum time is 30 minutes (1800 seconds) which is defined by the option `CloudLSUCheckInterval` in `spa.cfg`.

Troubleshooting the error when the bucket is created without msdpcldutil

Unlike the normal cloud LSU configuration, the bucket with Object Lock enabled cannot be created from NetBackup Web UI. You must use **msdpcldutil** to create the bucket with Object Lock enabled and create a cloud volume in it. If the bucket with Object Lock enabled already exists, you can use **msdpcldutil** to create a cloud volume in this bucket.

If you use AWS console or CLI to create a bucket directly instead of **msdpcldutil** tool, bucket loses the bucket policy protection, and you may see the following error. This issue is applicable only for NetBackup 9.1.

```
[root@rsvlmvc01vm linuxR_x86]# ./msdpcldutil create -b jzh-worm-bucket06
-v worm-b06-v02 --mode GOVERNANCE --min 1D --max 1Y -1 2023-10-24
current user has NO permission of cloud admin. Error: NoSuchBucketPolicy:
The bucket policy does not exist status code: 404, request id:
REQUESTID1234, host id: HostID1234
```

To fix this issue, you must add the bucket policy to S3 bucket manually. See “About bucket policy for immutable storage” on page 314.

See “About MSDP cloud admin tool” on page 307.

Deleting the immutable image with the governance mode

You can delete the immutable image with the governance mode retention lock.

To delete the immutable image with the governance 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.

   ```
   bpexpdate -backupid ${my_backup_id} -d 0 -try_expire_worm_copy
   -copy ${my_copy_num}
   ```

Refreshing the immutable cloud volume retention range value in the web UI

Sometimes, the cloud volume retention range value is not updated immediately on the NetBackup web UI after the configuration. If the value is not displayed correctly, run the following command, and refresh the web UI.

```
/usr/openv/netbackup/bin/admincmd/nbdevconfig -updatedp -dp
${diskpool_name} -stype PureDisk
```

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 an object-locked object (governance mode or expired) permanently, you must delete objects with their versioning.

About instant access for object storage in cloud

The following table describes the platforms supported by instant access for object storage in cloud.

Table 7-3

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 (Cloud Build-Your-Own, BYO-In-Cloud)	This platform is supported. You must manually enable this option.

The instant access for object storage in the cloud feature is enabled by default on the AKS/EKS platforms. For the cloud virtual machines to use instant access, you must manually perform the following steps to enable this feature:

1. Add the `instant-access-object-store = 1` option into the */etc/msdp-release* file on storage server.
2. On the master 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 master 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
   ```

Chapter 8

S3 Interface for MSDP

This chapter includes the following topics:

- About S3 interface for MSDP
- Prerequisites
- Configuring S3 interface for MSDP
- Identity and Access Management (IAM) for S3 interface for MSDP
- S3 APIs for S3 interface for MSDP
- 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 run on Build Your Own (BYO) environment (MSDP storage server).

Prerequisites

Following are the prerequisites to configure S3 interface for MSDP:

- The storage server operating system must be RHEL 7.6 or later.
- 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.
- Ensure that the **policycoreutils** and **policycoreutils-python** 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

After MSDP is configured, you can run `s3srv_config.sh` to configure S3 interface for MSDP.

To configure S3 server

- 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.
- 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.
  `systemctl status pdde-s3srv`
- After configuring S3 server, you can stop or start S3 server.

```
systemctl stop/start pdde-s3srv
```

- 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>
```

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.

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

Table 8-1 Common parameters *(continued)*

Parameters	Description
`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
`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

Table 8-1 Common parameters *(continued)*

Parameters	Description
`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
`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

Table 8-2 Common error codes *(continued)*

Error code	Description
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
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 339.

- `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 341.

- `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 339.

- `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 341.

- `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 339.

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 341.

- `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>1648203604905893069</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 339.

- `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 341.

- `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 339.

- `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 360.

Errors

For information about the errors that are common to all actions, See "Common Error Codes" on page 341.

- `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 339.

- `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 360.
- `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 341.

- `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>16484326126006469944</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 339.

- `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 341.

- `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 339.

- `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 341.

- `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 339.

- `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 341.

- `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>16484516123469945 99</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 339.

- `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 341.

- `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 339.

- `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 341.

- `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 339.

- `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 341.

- `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.

`s3:Put*`	S3 write operations.	`UploadPart`
		`CompleteMultipartUpload`
		`CreateMultipartUpload`
		`AbortMultipartUpload`
		`PutObject`
		`DeleteObject`
		`DeleteObjects`
		`PutBucketVersioning`
		`DeleteBucket`

`s3:Get*`	S3 read operations.	`HeadObject`
		`GetObject`
		`GetBucketVersioning`
		`GetBucketLocation`
		`GetBucketEncryption`
		`HeadBucket`

Table 8-4 Supported Action *(continued)*

Action	Description	Permissive APIs

`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.
`arn:aws:s3:::<BUCKET_NAME>/*`	All objects within <BUCKET_NAME>. And the bucket itself.

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>1653377472751453758</RequestId>
```

S3 APIs on Buckets

S3 APIs on buckets perform the following data arrangement functions:

- Create a bucket.
- Delete a bucket.
- Head a bucket.
- 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
```

Request Parameters

- `Bucket`
  Name of the bucket to be created.
  Required: Yes
  Type: String

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 returns a maximum of 1,000 multipart uploads in the response.

Request Syntax

```
GET /bucket?uploads&keymarker=KeyMarker&maxuploads
=MaxUploads&prefix=Prefix&upload-id-marker=UploadIdMarker
Host: msdps3.server:8443
```

Request Parameters

- `Bucket`
  Name of the bucket on which the multipart upload was initiated.
  Required: Yes
  Type: String
- `key-marker`
  The marker is the point where Object should begin listing objects. MSDP S3 expects the `key-marker` that was returned by server in last request. The value of `NextMarker` of response should be used in request as marker.
  Type: String
- `max-uploads`
  Limits the number of multipart uploads that are returned in the response. By default, the action returns up to 1,000 uploads.
  Type: Integer
- `prefix`
  Limits the response to uploads that begin with the specified prefix.
  Type: String
- `upload-id-marker`

With `key-marker`, you can specify an `upload-id-marker` after which listing should begin. If `key-marker` is not specified, `upload-id-marker` is ignored.
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. By default, the action returns up to 1,000 uploads.

- `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.

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

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.

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

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.

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

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.

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.

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.

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

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.

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

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

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.

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.

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 465.

See “Recovering from an MSDP storage server failure” on page 466.

You can recover cloud LSU in MSDP cloud. See “About the disaster recovery for cloud LSU” on page 281.

As for Scenario 1 and Scenario 2 (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.

Limitations in S3 interface for MSDP

S3 interface for MSDP has the following limitations:

- 1000 buckets in one S3 server.
- Recommend 50~100 concurrent requests based on BYO computer performance.
- Small file must be less than 1M. A tar compress file of small files must be less than 5G. The number of small files in small files tar compress file should be less than 10000.
- One object size must be less than 5TB.
- Multipart upload limitation. See Amazon S3 multipart upload limits
  Does not support Flex, Flex worm, FlexScale, and MSDP Scaleout.
- Does not support Amazon S3 Glacier, Deep Archive, and Microsoft Azure Archive.
- When you configure S3 server, the client's name in OST plug-in should avoid #s3storage.

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
- Viewing MSDP disk reports
- About monitoring MSDP processes
- Reporting on Auto Image Replication jobs

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 rate for a backup job in the Activity Monitor

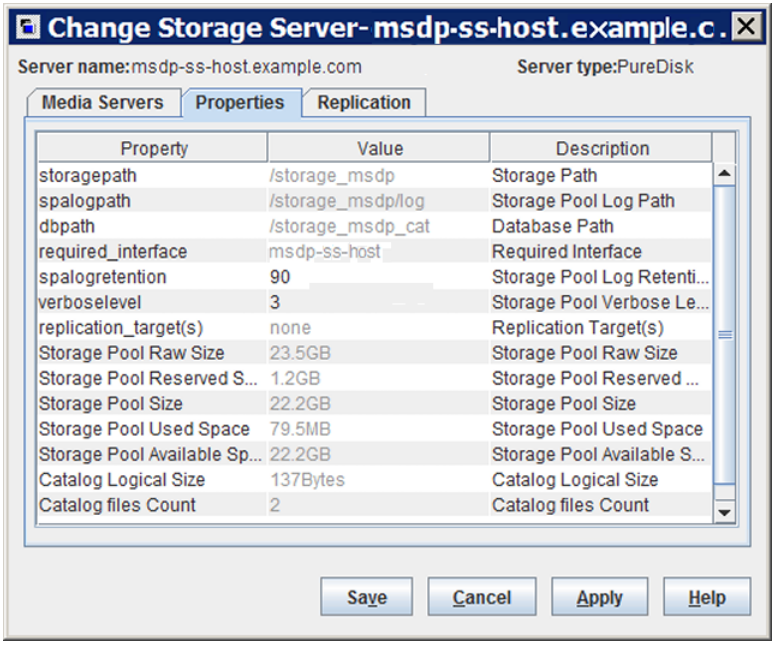
For the method to show the MSDP compression rate, See "Viewing MSDP job details" on page 411.

On UNIX and Linux, you can use the NetBackup `bpdbjobs` command to display the deduplication rate. However, you must configure it to do so.

See "To configure the `bpdbjobs` command to display the MSDP deduplication rate" on page 411.

To view the global MSDP deduplication ratio

1. In the NetBackup Administration Console, expand **Media and Device Management > Credentials > Storage Server**
2. Select the deduplication storage server.
3. On the **Edit** menu, select **Change**.
4. In the **Change Storage Server** dialog box, select the **Properties** tab. The **Deduplication Ratio** field displays the ratio.



To view the MSDP deduplication rate for a backup job in the Activity Monitor

1. In the **NetBackup Administration Console**, click **Activity Monitor**.
2. Click the **Jobs** tab.

   The **Deduplication Rate** column shows the rate for each job.

   Not all columns appear by default. Click **View > Column Layout** to show or hide columns.

To configure the `bpdbjobs` command to display the MSDP deduplication rate

- Add a `DEDUPRATIO BPDBJOBS_COLDEFS` entry in the `bp.conf` file on the media server on which you run the command.

  The `bpdbjobs` command then shows the deduplication rate in its output.

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.

Many factors affect deduplication performance.

See "About MSDP performance" on page 49.

Viewing MSDP job details

Use the NetBackup Activity Monitor to view deduplication job details.

To view MSDP job details

1. In the **NetBackup Administration Console**, 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 412.

MSDP job details

The **NetBackup Administration Console Job Details** dialog box shows 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 **Detailed Status** 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 **Detailed Status** 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 now 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 209.
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 72.
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 453.
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.

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 Administration Console. You can see an example of the dialog box in a different topic. See "Monitoring the MSDP deduplication and compression rates" on page 409.
Disk Pools window	The **Disk Pools** window of the Administration Console 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, expand **Media and Device Management > Devices > Disk Pools**.
View container command	A command that is installed with NetBackup provides a view of storage capacity and usage within the deduplication container files. See "About MSDP container files" on page 416. See "Viewing storage usage within MSDP container files" on page 416.
Disk Pool Status report	The **Disk Pool Status** report displays the state of the disk pool and usage information. See "Viewing MSDP disk reports" on page 417.

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 "Viewing MSDP disk reports" on page 417. See "MSDP event codes and messages" on page 588.
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 443.

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.

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%
                  1.0T  988.9G 666.0G  322.9G  68%

Number of containers             : 2981
Average container size           : 219740494 bytes (209.56MB)
Space allocated for containers   : 655046415189 bytes (610.06GB)
Reserved space                   : 45360705536 bytes (42.25GB)
Reserved space percentage        : 4.1%
```

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`.

Viewing MSDP disk reports

The NetBackup disk reports include information about the disk pools, disk storage units, disk logs, images that are stored on disk media, and storage capacity.

Table 9-3 describes the disk reports available.

Table 9-3 Disk reports

Report	Description
Images on Disk	The Images on Disk report generates the image list present on the disk storage units that are connected to the media server. The report is a subset of the Images on Media report; it shows only disk-specific columns. The report provides a summary of the storage unit contents. If a disk becomes bad or if a media server crashes, this report can let you know what data is lost.
Disk Logs	The Disk Logs report displays the media errors or the informational messages that are recorded in the NetBackup error catalog. The report is a subset of the Media Logs report; it shows only disk-specific columns. The report also includes information about deduplicated data integrity checking. See "About MSDP data integrity checking" on page 448. Either `PureDisk` or `Veritas Deduplication Engine` in the description identifies a deduplication message. The identifiers are generic because the deduplication engine does not know which application consumes its resources. NetBackup and Veritas Backup Exec are Veritas applications that use deduplication.
Disk Storage Unit Status	The Disk Storage Unit Status report displays the state of disk storage units in the current NetBackup configuration. For disk pool capacity, see the disk pools window in **Media and Device Management > Devices > Disk Pools**. Multiple storage units can point to the same disk pool. When the report query is by storage unit, the report counts the capacity of disk pool storage multiple times.
Disk Pool Status	The **Disk Pool Status** report displays the state of disk pool and usage information. This report displays only when a license that enables NetBackup disk features is installed.

To view disk reports

1. In the NetBackup Administration Console, expand **NetBackup Management > Reports > Disk Reports**.
2. Select the name of a disk report.

3 In the right pane, select the report settings.

4 Click **Run Report**.

About monitoring MSDP processes

The following table shows the deduplication processes about which NetBackup reports:

See "MSDP server components" on page 475.

Table 9-4 Where to monitor the main MSDP processes

What	Where to monitor it
NetBackup Deduplication Engine	In the **NetBackup Administration Console**, 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 Administration Console**, the NetBackup Deduplication Manager appears as `spad` on the **Daemons** tab of the **Activity Monitor**. The NetBackup `bpps` command also shows the `spad` process.

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 master server domain.

Table 9-5 Auto Image Replication jobs in the Activity Monitor

Job type	Description
Replication	The job that replicates a backup image to a target master 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 master 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 9-5 Auto Image Replication jobs in the Activity Monitor *(continued)*

Job type	Description
Import	The job that imports a backup copy into the target master 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 master. 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 master server. The image is expired and deleted during an **Image Cleanup** job. Note that the originating domain (Domain 1) does not track failed imports.

Chapter 10

Managing deduplication

This chapter includes the following topics:

- Managing MSDP servers
- Managing NetBackup Deduplication Engine credentials
- Managing Media Server Deduplication Pools
- Deleting backup images
- About MSDP queue processing
- Processing the MSDP transaction queue manually
- About MSDP data integrity checking
- Configuring MSDP data integrity checking behavior
- 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 master domain
- Specifying the restore server

Managing MSDP servers

After you configure deduplication, you can perform various tasks to manage deduplication servers.

See "Viewing MSDP storage servers" on page 422.

See "Determining the MSDP storage server state" on page 422.

See "Viewing MSDP storage server attributes" on page 423.

See "Setting MSDP storage server attributes" on page 424.

See "Changing MSDP storage server properties" on page 425.

See "Clearing MSDP storage server attributes" on page 426.

See "About changing the MSDP storage server name or storage path" on page 427.

See "Changing the MSDP storage server name or storage path" on page 427.

See "Removing an MSDP load balancing server" on page 429.

See "Deleting an MSDP storage server" on page 430.

See "Deleting the MSDP storage server configuration" on page 431.

Viewing MSDP storage servers

Use the NetBackup Administration Console to view a list of deduplication storage servers already configured.

To view MSDP storage servers

- In the NetBackup Administration Console, expand **Media and Device Management > Credentials > Storage Server**.

  The **All Storage Servers** pane shows all configured deduplication storage servers. Deduplication storage servers show **PureDisk** in the **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 master 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 master 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 436.

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 master 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 423.

See "About MSDP optimized synthetic backups" on page 48.

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

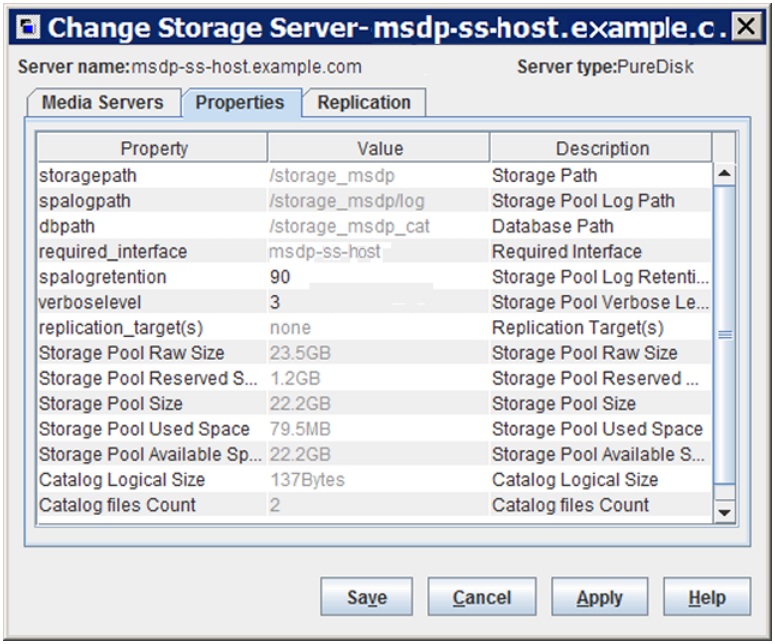
1 In the **NetBackup Administration Console**, expand **Media and Device Management > Credentials > Storage Servers**.

2 Select the deduplication storage server.

Warning: If you have load balancing servers, do not select any or all of them in the **Media Servers** pane of the **NetBackup Administration Console**. If you do, the change operation fails.

3 On the **Edit** menu, select **Change**.

4 In the **Change Storage Server** dialog box, select the **Properties** tab.



5 For the property to change, select the value in the **Value** column.

6 Change the value.

7 Click **OK**.

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 master server or on a storage server:

```
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 427.
- 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 427.
- 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 466.

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 427.

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 the *NetBackup Administrator's Guide, Volume I*: http://www.veritas.com/docs/DOC5332
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*: http://www.veritas.com/docs/DOC5332
Step 3	Delete the storage units that use the disk pool	See the *NetBackup Administrator's Guide, Volume I*: http://www.veritas.com/docs/DOC5332
Step 4	Delete the disk pool	See "Deleting a Media Server Deduplication Pool" on page 445.
Step 5	Delete the deduplication storage server	See "Deleting an MSDP storage server" on page 430.
Step 6	Delete the configuration	Delete the deduplication configuration. See "Deleting the MSDP storage server configuration" on page 431.
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 229.

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`

Table 10-1 Changing the storage server name or storage path *(continued)*

Step	Task	Procedure
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 54. See "MSDP storage path properties" on page 110.
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 MSDP server-side deduplication" on page 69.
Step 11	Import the backup images	See the *NetBackup Administrator's Guide, Volume I*: http://www.veritas.com/docs/DOC5332

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 39.

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 *NetBackup Administrator's Guide, Volume II*:

http://www.veritas.com/docs/DOC5332

To remove a media server from a MSDP node

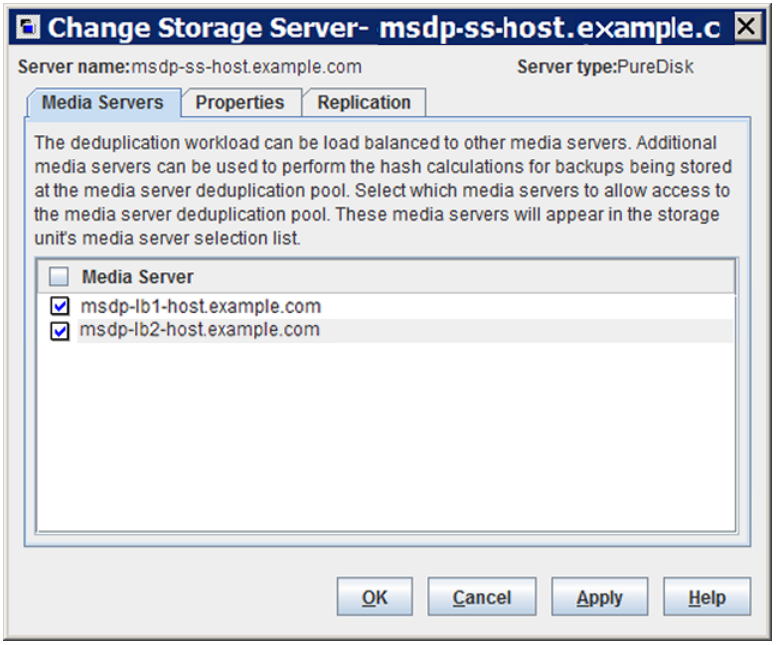
1. For every storage unit that specifies the media server in **Use one of the following media servers**, clear the check box that specifies the media server.

   This step is not required if the storage unit is configured to use any available media server.

2. In the NetBackup Administration Console, expand **Media and Device Management > Credentials > Storage Server**.

3 Select the deduplication storage server, and then select **Edit > Change**.

4 In the **Change Storage Server** dialog box, select the **Media Servers** tab.



5 Clear the check box of the media server you want to remove.

6 Click **OK**.

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. In the **NetBackup Administration Console**, expand **Media and Device Management > Credentials > Storage Server**
2. On the **Edit** menu, select **Delete**.
3. Click **Yes** in the confirmation dialog box.

See "Changing the MSDP storage server name or storage path" on page 427.

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 427.

See "Deactivating MSDP" on page 473.

To delete the MSDP storage server configuration

1. Use the NetBackup Administration Console 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 432.

See "Adding NetBackup Deduplication Engine credentials" on page 432.

See "Changing NetBackup Deduplication Engine credentials" on page 433.

See "Deleting credentials from a load balancing server" on page 433.

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. In the **NetBackup Administration Console**, expand **Media and Device Management > Credentials > Storage Server**.
2. Select the storage server, and then select **Edit > Change**.
3. In the **Change Storage Server** dialog box, select the **Media Servers** tab.

   The media servers for which credentials are configured are checked.

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 205.

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 Veritas support representative.

See "About the NetBackup Deduplication Engine credentials" on page 45.

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 429.

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 434.

See "Changing a Media Server Deduplication Pool properties" on page 437.

See "Determining the Media Server Deduplication Pool state" on page 434.

See "Changing OpenStorage disk pool state" on page 435.

See "Determining the MSDP disk volume state" on page 443.

See "Changing the MSDP disk volume state" on page 444.

See "Viewing Media Server Deduplication Pool attributes" on page 435.

See "Setting a Media Server Deduplication Pool attribute" on page 436.

See "Clearing a Media Server Deduplication Pool attribute" on page 442.

See "Resizing the MSDP storage partition" on page 456.

See "Inventorying a NetBackup disk pool" on page 444.

See "Deleting a Media Server Deduplication Pool" on page 445.

Viewing Media Server Deduplication Pools

Use the NetBackup Administration Console to view configured disk pools.

To view disk pools

- In the NetBackup Administration Console, expand **Media and Device Management > Devices > Disk Pools**.

Determining the Media Server Deduplication Pool state

The disk pool state is UP or DOWN.

To determine disk pool state

1. In the **NetBackup Administration Console**, expand **Media and Device Management > Device Monitor**.
2. Select the **Disk Pools** tab.
3. The state is displayed in the **Status** column.

Changing OpenStorage disk pool state

Disk pool state is UP or DOWN.

To change the state to DOWN, the disk pool 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.

Note: If you want to force the storage server to use other load balanced media server, deselect the storage server from the load balance media server list. Run the `bpstsinfo -resyncREM -servername <master-server-name` command to force a change.

For more information about using this issue, review this section:

Selecting a different media server to fix storage server and disk volume connectivity issues in NetBackup Administrator's Guide, Volume II.

To change OpenStorage disk pool state

1. In the **NetBackup Administration Console**, in the left pane, select **Media and Device Management > Device Monitor**.
2. At the bottom of the right pane, select the **Disk Pools** tab.
3. Select the disk pool.
4. Select either **Actions > Up** or **Actions > Down**.

Note: This wizard is also available in the NetBackup web UI. Click **Storage > Disk Pool**.

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 master 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.

See "Setting MSDP storage server attributes" on page 424.

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 master 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 435.

Changing a Media Server Deduplication Pool properties

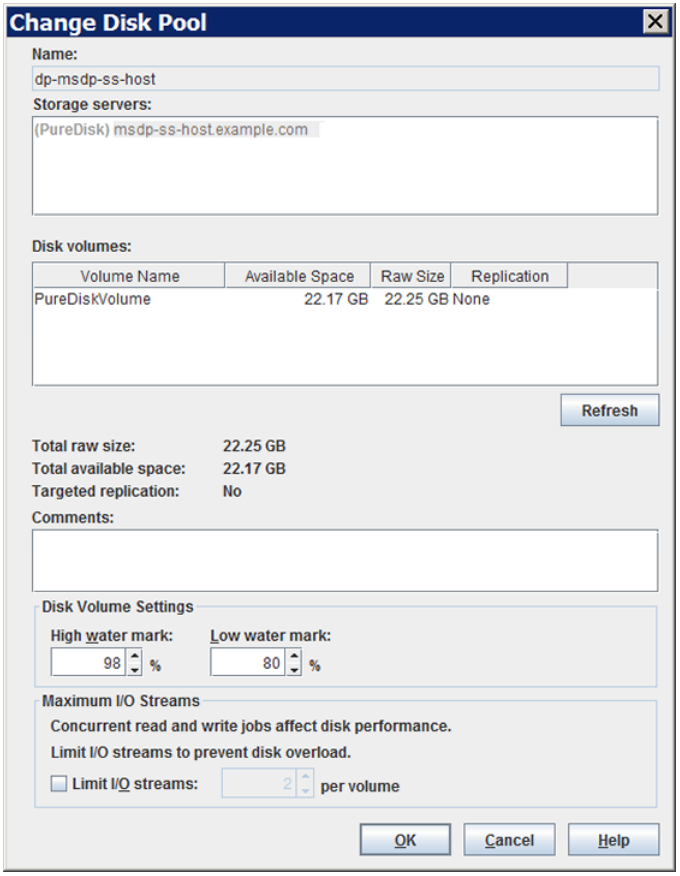
You can change the properties of a deduplication disk pool.

To change disk pool properties

1 In the NetBackup Administration Console, expand **Media and Device Management > Devices > Disk Pools**.

2 Select the disk pool you want to change in the details pane.

3 On the **Edit** menu, select **Change**.



4 In the **Change Disk Pool** dialog box, click **Refresh** to update the disk pool replication properties.

If NetBackup discovers changes, your actions depend on the changes discovered.

See "How to resolve volume changes for Auto Image Replication" on page 439.

5 Change the other properties as necessary.

See "Media Server Deduplication Pool properties" on page 116.

6 Click **OK**.

7 If you clicked **Refresh** and the **Replication** value for the **PureDiskVolume** changed, refresh the view in the **Administration Console**.

How to resolve volume changes for Auto Image Replication

When you open the **Change Disk Pool** dialog box, NetBackup loads the disk pool properties from the catalog. NetBackup queries the storage server for changes when you either click the **Refresh** button in the **Change Disk Pool** dialog box or when you configure a new disk pool for the storage server.

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 appear in the **Change Disk Pool** dialog box. Text in the dialog box changes to indicate that you can add the new volumes to the 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)*

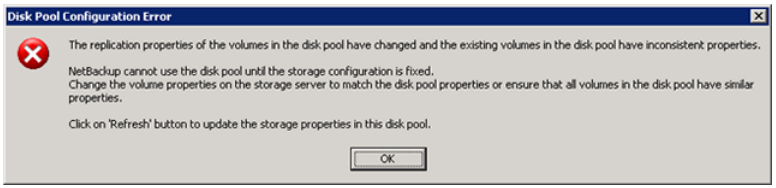
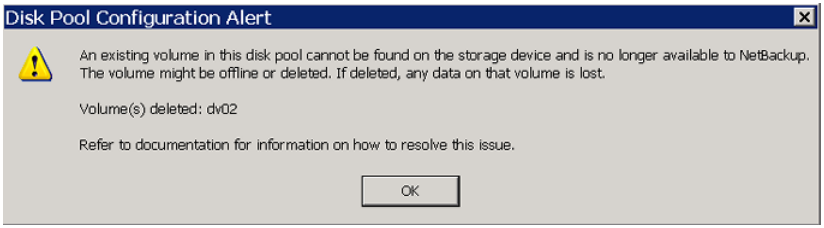
Outcome	Description
The replication properties of the volumes changed, and they are now inconsistent.	A **Disk Pool Configuration Error** pop-up box notifies you that the replication properties of some of the volumes in the disk pool changed. The properties of the volumes in the disk pool are not homogeneous.  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 163. 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 master 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
```

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**.

Inventorying a NetBackup disk pool

An inventory of a NetBackup disk pool reads the capacity of the disk volumes in the pool. An inventory operation lets you update NetBackup with the new capacity values if you do the following:

- Increase or decrease the size of the disk volumes in a disk pool.
- Add volumes to or remove volumes from a disk pool.

How you increase or decrease the underlying storage capacity depends on your storage implementation. You must complete that process before you inventory the disk pool.

To inventory a NetBackup disk pool

1. in the **NetBackup Administration Console**, select **Media and Device Management** > **Devices** > **Disk Pools**.
2. On the **Actions** menu, select **Inventory Disk Pools**.
3. In the **Inventory Disk Pool** dialog box, select the disk pool to inventory and then click **Start Inventory.**

4 To update the NetBackup catalog with the capacity values returned by the inventory, click **Update Configuration**.

5 To inventory another disk pool, go to step 3.

6 To exit, click **Close**.

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 581.

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 In the **NetBackup Administration Console**, expand **Media and Device Management > Devices > Disk Pools**.

2 Select a disk pool

3 On the **Edit** menu, select **Delete**.

4 In the **Delete Disk Pool** dialog box, verify that the disk pool is the one you want to delete and then click **OK**.

Deleting backup images

Image deletion may be time consuming. Therefore, if you delete images manually, Veritas recommends the following approach.

See "About the MSDP data removal process" on page 455.

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 567.

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 Veritas 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 446.

See "About MSDP server requirements" on page 40.

Processing the MSDP transaction queue manually

NetBackup maintains a queue for MSDP database transactions.

See “About MSDP queue processing” on page 446.

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 disd 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 disd 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 disd 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 Veritas Support. In such cases, NetBackup writes a message to the NetBackup Disk Logs report.

See “Viewing MSDP disk reports” on page 417.

The data integrity message code is 1057.

See “MSDP event codes and messages” on page 588.

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 567.

You can configure some of the data integrity checking behaviors.

See “Configuring MSDP data integrity checking behavior” on page 448.

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 449.
- Edit configuration file parameters.
  See "To configure data integrity checking behavior by editing the configuration files" on page 450.

Warning: Veritas 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 448.

See "MSDP data integrity checking configuration parameters" on page 451.

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 disd 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 451.

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 224.
- 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: Veritas 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 448.

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.

Table 10-3 The contentrouter.cfg file parameters for data integrity checking *(continued)*

Setting	Default	Description
ShutdownCRWhenError	false	Stops the NetBackup Deduplication Manager when a data loss is discovered. This parameter is reserved for debugging purposes by Veritas 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 10-4 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 567.

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 Veritas 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 453.

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 208.

See "MSDP pd.conf file parameters" on page 209.

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 10-5 describes the rebasing operations.

Table 10-5 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 209. 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.
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 209. 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 455.

MSDP server-side rebasing parameters

Table 10-6 describes the parameters that control server-side rebasing.

See "About MSDP storage rebasing" on page 453.

Usually, you do not need to change parameter values. However, in some cases, you may be directed to change settings by a Veritas support representative.

The parameters are stored in the `contentrouter.cfg` file.

See "About the MSDP contentrouter.cfg file" on page 224.

Table 10-6 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 416.

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 448.

See "Deleting backup images" on page 445.

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*:

   http://www.veritas.com/docs/DOC5332
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*:

http://www.veritas.com/docs/DOC5332

7 Restart the deduplication jobs.

How MSDP restores work

The following two methods exist to for MSDP restore operations:

Table 10-7 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 40. ■ 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 143. You can specify the server to use for restores. See "Specifying the restore server" on page 460.

Table 10-7 MSDP restore types *(continued)*

Type	Description
Restore directly to the client	The storage server can bypass the media server and move the data directly to the client. NetBackup does not choose a media server for the restore, and the restore does not use the media server `bptm` process. 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 458. By default, NetBackup rehydrates the data on the storage server except for client-side deduplication clients. Those clients rehydrate the data. You can configure NetBackup so that the data is rehydrated on the storage server rather than the client. See the `RESTORE_DECRYPT_LOCAL` parameter in the MSDP `pd.conf` file. See "MSDP pd.conf file parameters" on page 209. See "Editing the MSDP pd.conf file" on page 209.

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 457.

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 154.

2. On the master 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*:

http://www.veritas.com/docs/DOC5332

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 master domain

While it is possible to restore a client directly by using the images in the target master 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 10-8 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 457.

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*:
    http://www.veritas.com/docs/DOC5332
    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 master 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*:

http://www.veritas.com/docs/DOC5332

Chapter 11

Recovering MSDP

This chapter includes the following topics:

- About recovering the MSDP catalog
- Restoring the MSDP catalog from a shadow copy
- Recovering from an MSDP storage server disk failure
- Recovering from an MSDP storage server failure
- Recovering the MSDP storage server after NetBackup catalog recovery

About recovering the MSDP catalog

The following are the recovery options for the NetBackup MSDP catalog:

Table 11-1 MSDP catalog backup recovery options

Recovery option	Description
Restore from a shadow copy	If NetBackup detects corruption in the MSDP catalog, the Deduplication Manager restores the catalog automatically from the most recent shadow copy. The automatic restore process also plays a transaction log so that the recovered MSDP catalog is current. Although the shadow copy restore process is automatic, a restore procedure is available if you need to recover from a shadow copy manually. See "About the MSDP shadow catalog" on page 231. See "Restoring the MSDP catalog from a shadow copy" on page 463.

Table 11-1 MSDP catalog backup recovery options *(continued)*

Recovery option	Description
Recover from a backup	If you configured an MSDP catalog backup policy and a valid backup exists, you can recover the catalog from a backup. As a general rule, you should only attempt to recover the MSDP catalog from a backup if you have no alternatives. As an example: A hardware problem or a software problem results in the complete loss of the MSDP catalog and the shadow copies. The greatest chance for a successful outcome when you recover the MSDP catalog from a backup is when the recovery is guided. An unsuccessful outcome may cause data loss. For the customers who need to recover the MSDP catalog, Veritas wants to guide them through the process. Therefore, to recover the MSDP catalog from a backup, contact your Veritas support representative. You can refer the support representative to Knowledge Base Article 000047346, which contains the recovery instructions.

Caution: You must determine if your situation is severe enough to recover the catalog. Veritas recommends that you contact your Veritas Support representative before you restore or recover the MSDP catalog. The Support representative can help you determine if you need to recover the catalog or if other solutions are available.

See "About protecting the MSDP catalog" on page 231.

Restoring the MSDP catalog from a shadow copy

NetBackup automatically restores the necessary parts of the MSDP catalog if corruption is detected. However, you can restore the MSDP catalog from a shadow copy manually, although in normal circumstances it is not necessary. Veritas recommends that you contact your Veritas Support representative before you restore all or part of the MSDP catalog from a shadow copy.

The procedure that you use depends on the restore scenario, as follows:

Restore the entire MSDP catalog from a shadow copy	In this scenario, you want to restore the entire catalog from one of the shadow copies. See "To restore the entire MSDP catalog from a shadow copy" on page 464.

Restore a specific MSDP database file	The MSDP catalog is composed of multiple small database files. Those files are organized in the file system by the client name and policy name, as follows: UNIX: */database_path*/databases/catalogshadow/2/*ClientName*/*PolicyName* Windows: *database_path*\databases\catalogshadow\2*ClientName**PolicyName* You can restore the database files for a client and a policy combination. The restore of a specific client's and policy's database files is always from the most recent shadow copy. See "To restore a specific MSDP database file from a shadow copy" on page 464.

See "About recovering the MSDP catalog" on page 462.

To restore the entire MSDP catalog from a shadow copy

1. If any MSDP jobs are active, either cancel them or wait until they complete.
2. Disable all policies and storage lifecycle policies that back up to the **Media Server Deduplication Pool**.
3. On the MSDP storage server, run the following command, depending on host type:
   - UNIX: `/usr/openv/pdde/pdcr/bin/cacontrol --catalog recover all`
   - Windows: *install_path*`\Veritas\pdde\cacontrol --catalog recover all`
4. Enable all policies and storage lifecycle policies that back up to the **Media Server Deduplication Pool**.
5. Restart the jobs that were canceled before the recovery.

To restore a specific MSDP database file from a shadow copy

1. If any MSDP jobs are active for the client and the backup policy combination, either cancel them or wait until they complete.
2. Disable the policies and storage lifecycle policies for the client and the backup policy combination that back up to the **Media Server Deduplication Pool**.

3. Change to the shadow directory for the client and policy from which you want to recover that database file. That directory contains the database files from which to recover. The following are the pathname formats:

   UNIX:
   `/database_path/databases/catalogshadow/2/ClientName/PolicyName`

   Windows:
   `database_path\databases\catalogshadow\2\ClientName\PolicyName`

4. Run the following command, depending on host type:

   - UNIX: `/usr/openv/pdde/pdcr/bin/cacontrol --catalog recover 2 "/ClientName/PolicyName"`
   - Windows: `install_path\Veritas\pdde\cacontrol --catalog recover 2 "\ClientName\PolicyName"`

5. Enable all policies and storage lifecycle policies that back up to the **Media Server Deduplication Pool**.

6. If you canceled jobs before you recovered the database files, restart them.

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.

Veritas 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 11-2 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.
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*: http://www.veritas.com/docs/DOC5332 See "About the MSDP license" on page 64.
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 229.
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 433.
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 432.
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 226.
Step 8	Edit the configuration file	See "Editing an MSDP storage server configuration file" on page 227.
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 228.
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 205.

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.

- Veritas recommends that the new computer use the same operating system as the old computer.
- Veritas 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 Veritas support representative.

Table 11-3 Recover from an MSDP storage server failure

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*: http://www.veritas.com/docs/DOC5332
Step 2	Delete the storage units that use the disk pool	See the *NetBackup Administrator's Guide, Volume I*: http://www.veritas.com/docs/DOC5332
Step 3	Delete the disk pool	See "Deleting a Media Server Deduplication Pool" on page 445.
Step 4	Delete the deduplication storage server	See "Deleting an MSDP storage server" on page 430.

Table 11-3 Recover from an MSDP storage server failure *(continued)*

Step	Task	Procedure
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 229.
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 433.

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 39.
		
		See "About MSDP server requirements" on page 40.

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 for UNIX and Windows*: http://www.veritas.com/docs/DOC5332

Table 11-3 Recover from an MSDP storage server failure *(continued)*

Step	Task	Procedure
Step 10	Reconfigure deduplication	You must use the same credentials for the NetBackup Deduplication Engine. See "Configuring MSDP server-side deduplication" on page 69.
Step 11	Import the backup images	See the *NetBackup Administrator's Guide, Volume I*: http://www.veritas.com/docs/DOC5332

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 228.

Veritas recommends that you save your storage server configuration.

See "Save the MSDP storage server configuration" on page 59.

Information about recovering the master server is available.

See *NetBackup Troubleshooting Guide*:

http://www.veritas.com/docs/DOC5332

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.

- Veritas recommends that the new computer use the same operating system as the old computer.
- Veritas 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 Veritas 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*: http://www.veritas.com/docs/DOC5332
Step 2	Delete the storage units that use the disk pool	See the *NetBackup Administrator's Guide, Volume I*: http://www.veritas.com/docs/DOC5332
Step 3	Delete the disk pool	See "Deleting a Media Server Deduplication Pool" on page 445.
Step 4	Delete the deduplication storage server	See "Deleting an MSDP storage server" on page 430.
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 229.
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 433.

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 39.
		See "About MSDP server requirements" on page 40.

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 for UNIX and Windows*: http://www.veritas.com/docs/DOC5332
Step 10	Reconfigure deduplication	See "Configuring MSDP server-side deduplication" on page 69.
Step 11	Import the backup images	See the *NetBackup Administrator's Guide, Volume I*: http://www.veritas.com/docs/DOC5332

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 427.
- Deactivate deduplication and remove the configuration files and the storage files.
  See "Deactivating MSDP" on page 473.

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 127.
Step 2	Delete the storage units that use the disk pool	See the *NetBackup Administrator's Guide, Volume I*: http://www.veritas.com/docs/DOC5332
Step 3	Delete the disk pool	See "Deleting a Media Server Deduplication Pool" on page 445.
Step 4	Delete the deduplication storage server	See "Deleting an MSDP storage server" on page 430. 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 431.
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 229.
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

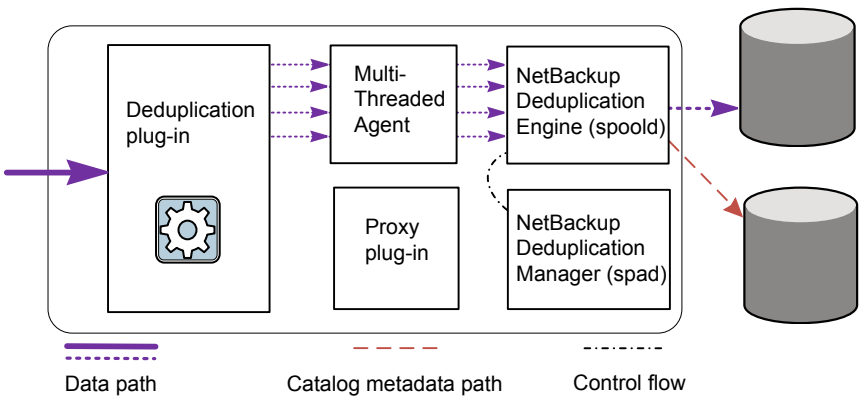


Table 14-1 describes the MSDP server components.

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 72.
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 Administration Console.
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 Administration Console.
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 448.
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 416.
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 81.
Queue processing	The NetBackup Deduplication Engine processes the transaction queue. See "About MSDP queue processing" on page 446.
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.

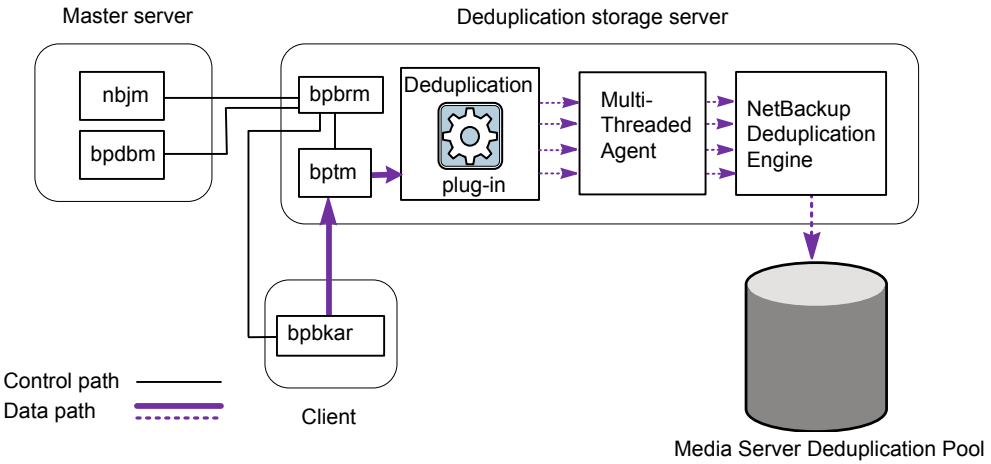
Table 14-2 NetBackup Deduplication Engine components and functions *(continued)*

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 master 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.

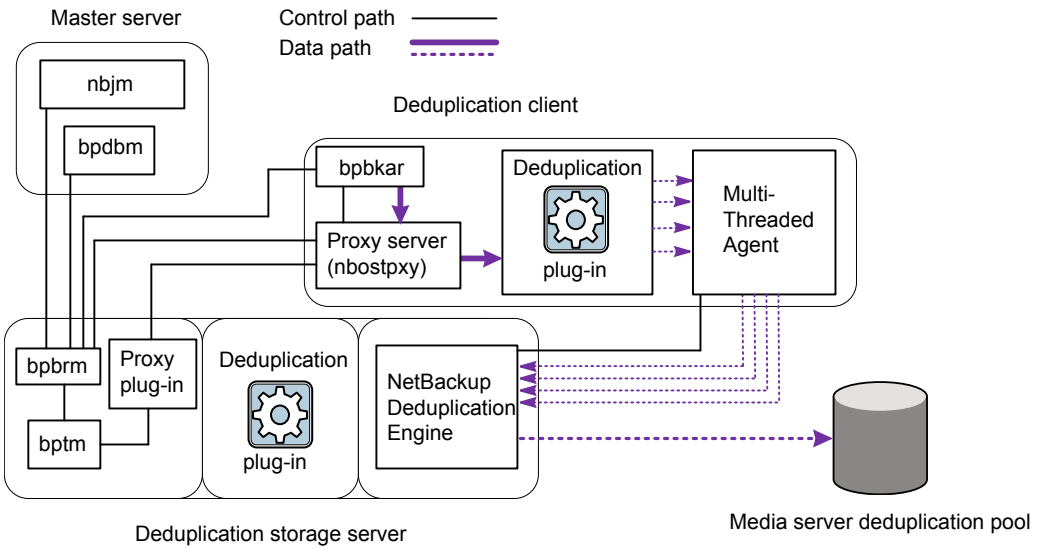
Table 14-3 Client MSDP components *(continued)*

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 72.
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 master 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 83.
  - 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 83.

  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 using universal shares

This chapter includes the following topics:

- About universal shares
- Configuring and using an MSDP build-your-own (BYO) server for universal shares
- MSDP build-your-own (BYO) server prerequisites and hardware requirements to configure universal shares
- Configuring universal share user authentication
- Mounting a universal share created from the NetBackup web UI
- Creating a protection point for a universal share
- About universal share self-service recovery
- Performing a universal share self-service recovery
- Using the ingest mode
- About universal shares with object store
- Enabling a universal share with object store
- Disaster recovery for a universal share
- Changing the number of vpfsd instances
- Upgrading to NetBackup 10.1

About universal shares

The universal share feature provides data ingest into an existing NetBackup deduplication pool (MSDP) or a supported Veritas appliance using an NFS or a CIFS (SMB) share. Space efficiency is achieved by storing this data directly into an existing NetBackup-based Media Server Deduplication Pool.

Advantages of universal shares

The following information provides a brief description of the advantages for using universal shares:

- As a 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.
- As a 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 CPD 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

Client software is not required for universal share backups or restores. Universal shares work with any POSIX-compliant operating system that supports NFS or CIFS.

How it works

The universal share feature provides a network-attached storage (NAS) option for supported Veritas 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 appliance, 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.

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 or through the NetBackup Administration Console.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.

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 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.

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 "MSDP build-your-own (BYO) server prerequisites and hardware requirements to configure universal shares" on page 489..
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 created from the NetBackup web UI" on page 495.
4	Configure a universal share backup policy. See "Creating a protection point for a universal share" on page 497.
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 "Using the ingest mode" on page 500.

Table 15-1 Process for configuring and using universal shares with an MSDP build-your-own (BYO) server *(continued)*

Step	Description
6	Restore from a universal share backup. Besides offering a fast data protection process, the Protection Point offers two powerful restore methods: Client-based restore: ■ Data protected using a Protection Point (see step 4 in this table) 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 master server (`https://<primary_server>/api-docs/index.html`). It can also be found online.

MSDP 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, 7.7, 7.8, 7.9, 8.1, 8.2, and 8.3.
- You must set up user authentication for the universal share.
  See "Configuring universal share user authentication" on page 491.
- NFS services must be installed and running if you want to use the share over NFS.
- Samba services must be installed and running if you want to use share over CIFS/SMB.
  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 491.
- Ensure that the `nfs-utils` is installed:
  - `yum install nfs-utils -y`
- Ensure that the Linux `samba` and `samba winbind` packages are installed.
  - `yum install samba-common samba-winbind samba-winbind-clients samba-winbind-modules -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`
- NGINX is installed and running.
  - Installing 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.)

- Installing NGINX from the EPEL repository:
  - Refer to https://fedoraproject.org/wiki/EPEL for installation instructions of the repository and further information.
    The EPEL repository is a volunteer-based community effort and not commercially supported by Red Hat.
- Before you start the storage configuration, ensure that the new BYO NGINX configuration entry `/etc/nginx/conf.d/byo.conf` is included as part of the HTTP section of the original `/etc/nginx/nginx.conf` file.
- If SE Linux has been configured, ensure that the `policycoreutils` and `policycoreutils-python` packages are installed from the same RHEL yum source (RHEL server), and then run the following commands:
  - `semanage port -a -t http_port_t -p tcp 10087`
  - `setsebool -P httpd_can_network_connect 1`

  Enable the `logrotate` permission in SE Linux using the following command:

  `semanage permissive -a logrotate_t`

- 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.

If you configure the universal share feature on BYO after storage is configured or upgraded without the NGINX service installed, run the command:

```
/usr/openv/pdde/vpfs/bin/vpfs_config.sh --configure_byo
```

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).

Configuring universal share user authentication

The universal share created with CIFS/SMB protocol supports two methods of user authentication:

- Active Directory-based user authentication
- Local user-based authentication

Active Directory-based authentication

If the appliance, 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.

For information about setting up Active Directory users or groups with a Flex Appliance application instance, see the NetBackup 10.1 Application Guide for Flex Appliance OS.

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.
Flex media server HA	The same as BYO.

WORM enabled storage server	The storage server can be configured to join or leave Active Directory with Restricted Shell commands.
	
	`[msdp-16.0] deecl01vm046p3 >`
	`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 526.

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 checked using the `wbinfo` command to ensure that they are valid. The following procedure describes how to add a universal share to an Active Directory.

Adding a universal share to an Active Directory

1. Create a universal share with SMB protocol on NetBackup web UI.
2. Mount the shared storage on a Windows client.

   Provide all necessary credentials.
3. Verify that the universal share is fully set up, and can be backed up and restored using a **Universal-Share** policy.

The following items must be set up for to add Microsoft SQL Instant Access to an Active Directory:

- Storage server and client must be in the same domain.
- Use domain user to log in to the Microsoft SQL client.
- Register Microsoft SQL instance with the domain user on the NetBackup Administration Console.
- Must use domain user credentials to do 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.

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 `test_smb_user use` for the SMB service only:

```
# adduser --no-create-home -s /sbin/nologin test_smb_user
# smbpasswd -a test_smb_user
```

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 RShell in the MSDP engine.
    ```
    su - msdpadm
    ```
  - Run the following RShell command to add a SMB user.
    ```
    setting samba add-user username=[samba user name]
    password=[samba password]
    ```
    For example,
    ```
    msdp-16.1] > setting samba add-user username=test_samba_user
    password=Te@Pss1fg0
    ```

You can use the same command to update the password for an existing user.

In AKS and EKS cloud platforms, the SMB RShell command configures SMB servers in all MSDP engines in a cluster.

Mounting a universal share created from the NetBackup web UI

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 > Storage Configuration > Universal Share**
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 > Storage Configuration > Universal Share**

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
     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.
     ```

You can find the mount path on the NetBackup web UI: **Storage > Storage Configuration > Universal Share**.

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 servers should be configured with its own specific policy to protect the universal shares on that storage server.

More information is available:

See "About universal shares" on page 484.

To create a protection point policy for a universal share

1. Create a policy using with the NetBackup Administration Console or 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.

Note: If primary server or MSDP storage server is running NetBackup 10.0.1 or later, media server must also be 10.0.1 or later.

---

4 On the **Schedules** tab, select either **FULL** or **INCR**.

Note: Accelerator backups are not supported or necessary for universal shares.

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 **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 /demo/database1 USING /mnt/vpfs_shares/3cc7/3cc77559-64f8-4ceb-be90-3e242b89f5e9`. In this example, the backup will be 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 in cloud" on page 332.

For information about NetBackup APIs, see the following website:

https://sort.veritas.com/documents

Select NetBackup and then the version at the bottom of the page.

About universal share self-service recovery

This feature allows a workload administrator to restore universal share data using Instant Access or single file recovery in the 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 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**: NSF 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.

Using 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_path>/.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_path>/.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.

Using the ingest mode to take a 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

1. Take the snapshot from the client side.
2. 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
   ```

Using the ingest mode to run a policy using 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
```

About universal shares with 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 15-4

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 503.

Note: Ensure that you create a new copy of the `auth.key` file for each media server and or node on a separate computer when created and each time it is changed (Example: after a regular MSDP disaster recovery). The file is required to perform a universal share disaster recovery.

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 instance 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 master 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.

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]
```

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 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
   ```

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.

Disaster recovery for a universal share

Universal share disaster recovery is available for BYO, AKS, and EKS environments when data in a share has been corrupted and 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 hostname of the computer the disaster recovery is performed on must match the hostname the shares were originally created on. Also, you must make a copy of the `auth.key` file because the file is required for the disaster recovery.

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 a disaster recovery for all universal shares

1. Navigate to the following location on the media server:

   `/usr/openv/pdde/vpfs/bin`

2. Ensure that the `auth.key` file in your storage path is the same as when a share was last created or deleted. The file contents may be different if you perform a universal share disaster recovery after a regular MSDP disaster recovery.
3. Run the following:

   `./vpfs_actions -a disasterRecovery --cloudVolume` *`CLOUDVOLUMENAME`*

   (where `cloudVolume` is the name of the MSDP cloud volume)
4. Revert your `auth.key` file if you replaced it in step 2.
5. NetBackup automatically performs the following:
   - Downloads all of the share scripts from the MSDP cloud volume bucket except for `vpfs0.sh` which should be recovered during MSDP disaster recovery. NetBackup also adds executable permissions to the scripts.
   - Downloads the NFS export list (if it exists) from the cloud.
   - Downloads the SMB export list (if it exists) from the cloud.
   - NetBackup calls the `vpfsadm --ioctlProxy` command to recreate the shares locally.
   - NetBackup mounts the shares for recovery.
     - Restart the NetBackup server if it's BYO.
       - Stop all NetBackup services using `/usr/openv/netbackup/bin/goodies/netbackup stop`.
       - Start all NetBackup services using `/usr/openv/netbackup/bin/goodies/netbackup start`.
     - Restart the MSDP node if it's in the MSDP cluster.

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 1 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 1 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
   ```

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
```

Upgrading to NetBackup 10.1

You must unmount all the NFS mount points on the client side before you upgrade from a previous release to NetBackup 10.1. Otherwise, problems can occur 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 on the Linux UNIX client.
2. Upgrade to NetBackup 10.1.
3. Start the NetBackup services.
4. Mount the universal share on the Linux UNIX client.

Chapter 16

Configuring isolated recovery environment (IRE)

This chapter includes the following topics:

- Configuring an isolated recovery environment on a NetBackup BYO media server
- Managing an isolated recovery environment on a NetBackup BYO media server
- Configuring AIR for replicating backup images from production environment to IRE BYO environment

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 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 arget primary server username
   ```

   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 *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:

   ```
   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 1:00 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.

---

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`*

  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 *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:

    ```
    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.

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
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 AIR 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: AIR 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 AIR for replicating backup images from production environment to IRE BYO environment, ensure the following:

- In 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 AIR for replicating backup images from production environment to IRE BYO environment

1. Run the following commands:

   ```
   /usr/openv/netbackup/bin/nbcertcmd -getCACertificate -server
   production primary server>
   ```

   ```
   /usr/openv/netbackup/bin/nbcertcmd -getCertificate -server
   production primary server> -token token
   ```

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 ]
```

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 *slp_window_name* is the name of the SLP window to be synched 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 WORM 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 WORM 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]
```

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 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 *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 WORM storage server.

The *target storage server username* is the username of the target WORM storage server.

The *slp_window_name* is the name of the SLP window that is synched 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.

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
- Managing certificates from the deduplication shell
- Managing FIPS 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

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.
  See "About the support command" on page 547.

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
- All other products:
  - Username: **msdpadm**
  - Password: **P@ssw0rd**

  You are required to change your password the first time you log in.

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 524.

- MSDP users

  See "Adding MSDP users from the deduplication shell" on page 525.
- 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 526.

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> password=<password>
   ```

   Where *<username>* is the username of the user that you want to add, and *<password>* is a password for that 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:

   ```
   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 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 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.

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
   - 145
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 the following command:

```
setting MSDP-user reset-password
```

4 Follow the prompt to enter the new 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 (_./@*!%#).

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>
```

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 252.

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 the following command:

   ```
   retention policy disable backup_ID=<ID> copynumber=<number>
   ```

   You can find the backup ID and the copy number in the output of the `retention policy list` command.

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 252.

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
   ```

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*> is the passphrase of the private key 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*> is the passphrase of the private key 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
```

Encrypting backups from the deduplication shell

To encrypt backups on a WORM or an MSDP storage server, you can configure the Key Management Service (KMS) or MSDP encryption.

Note: Do not use KMS and MSDP encryption together.

Use the following procedures to configure encryption for your backups from the deduplication shell.

To configure KMS encryption

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 hostname 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

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.

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`

Parameter	Description	Commands
`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-fp-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 539. 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 539. 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`

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 541.
- The content router queue processing (CRQP) service

See "Managing the content router queue processing (CRQP) service" on page 542.

- The online checking service
  See "Managing the online checking service" on page 543.
- The compaction service
  See "Managing the compaction service" on page 543.
- The deduplication (MSDP) services
  See "Managing the deduplication (MSDP) services" on page 544.
- The Storage Platform Web Service (SPWS)
  See "Managing the Storage Platform Web Service (SPWS)" on page 545.
- The Veritas provisioning file system (VPFS) mounts
  See "Managing the Veritas provisioning file system (VPFS) mounts" on page 545.
- The NGINX service
  See "Managing the NGINX service" on page 546.
- The SMB service
  See "Managing the SMB service" on page 547.
- The following deduplication utilities:
  - The deduplication manager utility (`cacontrol`)
    See "Oracle stream handler" on page 51.
  - The deduplication engine utility (`crcontrol`)
    See "Viewing storage usage within MSDP container files" on page 416.

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 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 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
   ```

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 start
```

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-self-signed-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 SPWS

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
```

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 `support` command
  See "About the support command" on page 547.
- 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 279.

About the support command

The `support` command lets you access and upload the relevant logs and configuration files for troubleshooting.

The following table describes the options and arguments for the `support` command.

Table 17-1 The options and arguments for the `support` command.

Option and its description	Argument	Description
`MSDP-history` Access the MSDP history files.	`tail`	Append the last 10 lines of each file to standard output. This command requires the following parameters: ■ `file` Enter the filename. This is a required parameter. ■ `options` Enter the supported options for `grep: -i, -w, -E`, or `tail: -f, -n`.
	`collect`	Collect files for transferring to the target host. This command requires the following parameters: ■ `mmin` Find files that are modified `n` minutes ago. Same as `mmin` option in find command. For example, 15, +15, -15. ■ `pattern` Specify the naming pattern to find and select matching files and folders. For example, `spoold`, or `spad*`. ■ `mtime` Finds files that are modified `n*24` hours ago. Same as `mtime` option in find command. For example, 2, +2, -2.
	`ls`	List information about the FILEs. Use the `dir` parameter to enter the directory name.
	`scp`	

Table 17-1 The options and arguments for the `support` command. *(continued)*

Option and its description	Argument	Description
		Securely transfer selected files to the target host. This command requires the following parameters: ■ `scp_port` Specifies the port to connect to on the remote host. ■ `scp_target` Specifies the target host and path for transferring files. Use the `user@host:/directory path/` format. This is a required parameter.
	`cat`	Concatenate files and print on the standard output. Use the `file` parameter to enter the file name.
	`grep`	Print lines that match patterns. This command requires the following parameters: ■ `pattern` Specify the naming pattern to find and select matching files and folders. For example, `spoold`, or `spad*`. This is a required parameter. ■ `file` Enter the filename. This is a required parameter. ■ `options` Enter the supported options for `grep: -i, -w, -E`, or `tail: -f, -n`.

Table 17-1 The options and arguments for the `support` command. *(continued)*

Option and its description	Argument	Description
`process` Displays information about the MSDP processes.	`MSDP-process`	Display the MSDP processes.
	`htop`	Display the CPU and memory information.
	`atop`	Display information about the operating system.
	`pidstat`	Display the PID statistics.
	`memory-usage`	Displays the free and used memory.
`software` Displays information about the software.	`show-MSDP-version`	Display the MSDP version.
	`show-OS-version`	Display the operating system information.

Table 17-1 The options and arguments for the `support` command. *(continued)*

Option and its description	Argument	Description
`MSDP-log` Access the MSDP log files.	`tail`	Append the last 10 lines of each file to standard output. This command requires the following parameters: ■ `file` Enter the filename. This is a required parameter. ■ `options` Enter the supported options for `grep: -i, -w, -E`, or `tail: -f, -n`.
	`collect`	Collect files for transferring to the target host. This command requires the following parameters: ■ `mmin` Find files that are modified `n` minutes ago. Same as `mmin` option in find command. For example, 15, +15, -15. ■ `pattern` Specify the naming pattern to find and select matching files and folders. For example, `spoold`, or `spad*`. ■ `mtime` Finds files that are modified `n*24` hours ago. Same as `mtime` option in find command. For example, 2, +2, -2.
	`ls`	List information about the files. Use the `dir` parameter to enter the directory name.
	`scp`	

Table 17-1 The options and arguments for the `support` command. *(continued)*

Option and its description	Argument	Description
		Securely transfer selected files to the target host. This command requires the following parameters: ■ `scp_port` Specifies the port to connect to on the remote host. ■ `scp_target` Specifies the target host and path for transferring files. Use the `user@host:/directory path/` format. This is a required parameter.
	`cat`	Concatenate files and print on the standard output. Use the `file` parameter to enter the file name.
	`grep`	Print lines that match patterns. This command requires the following parameters: ■ `pattern` Specify the naming pattern to find and select matching files and folders. For example, `spoold`, or `spad*`. This is a required parameter. ■ `file` Enter the filename. This is a required parameter. ■ `options` Enter the supported options for `grep: -i, -w, -E`, or `tail: -f, -n`.
`hardware` Displays information about the hardware.	`cpumem`	Display the CPU and memory information.

Table 17-1 The options and arguments for the `support` command. *(continued)*

Option and its description	Argument	Description
`MSDP-config` Access the MSDP configuration files.	`tail`	Append the last 10 lines of each file to standard output. This command requires the following parameters: ■ `file` Enter the filename. This is a required parameter. ■ `options` Enter the supported options for `grep: -i, -w, -E`, or `tail: -f, -n`.
	`collect`	Collect files for transferring to the target host. This command requires the following parameters: ■ `mmin` Find files that are modified `n` minutes ago. Same as `mmin` option in find command. For example, 15, +15, -15. ■ `pattern` Specify the naming pattern to find and select matching files and folders. For example, `spoold`, or `spad*`. ■ `mtime` Finds files that are modified `n*24` hours ago. Same as `mtime` option in find command. For example, 2, +2, -2.
	`ls`	List information about the FILEs. Use the `dir` parameter to enter the directory name.
	`scp`	

Table 17-1 The options and arguments for the `support` command. *(continued)*

Option and its description	Argument	Description
		Securely transfer selected files to the target host. This command requires the following parameters: ■ `scp_port` Specifies the port to connect to on the remote host. ■ `scp_target` Specifies the target host and path for transferring files. Use the `user@host:/directory path/` format. This is a required parameter.
	`cat`	Concatenate files and print on the standard output. Use the `file` parameter to enter the file name.
	`grep`	Print lines that match patterns. This command requires the following parameters: ■ `pattern` Specify the naming pattern to find and select matching files and folders. For example, `spoold`, or `spad*`. This is a required parameter. ■ `file` Enter the filename. This is a required parameter. ■ `options` Enter the supported options for `grep: -i, -w, -E`, or `tail: -f, -n`.

Table 17-1 The options and arguments for the `support` command. *(continued)*

Option and its description	Argument	Description
`diskio` Displays information about the disk I/O.	`iostat`	Displays the information about the disk I/O.
	`vmstat`	Displays the information on the wait on the disk I/O.
	`nmon`	Display the information about the monitor system.
	`disk-volume`	Display information about the disk volume.

Table 17-1 The options and arguments for the `support` command. *(continued)*

Option and its description	Argument	Description
`syslogs` Access the system logs.	`tail`	Append the last 10 lines of each file to standard output. This command requires the following parameters: ■ `file` Enter the filename. This is a required parameter. ■ `options` Enter the supported options for `grep: -i, -w, -E`, or `tail: -f, -n`.
	`collect`	Collect files for transferring to the target host. This command requires the following parameters: ■ `mmin` Find files that are modified `n` minutes ago. Same as `mmin` option in find command. For example, 15, +15, -15. ■ `pattern` Specify the naming pattern to find and select matching files and folders. For example, `spoold`, or `spad*`. ■ `mtime` Finds files that are modified `n*24` hours ago. Same as `mtime` option in find command. For example, 2, +2, -2.
	`ls`	List information about the FILEs. Use the `dir` parameter to enter the directory name.
	`scp`	

Table 17-1 The options and arguments for the `support` command. *(continued)*

Option and its description	Argument	Description
		Securely transfer selected files to the target host. This command requires the following parameters: ■ `scp_port` Specifies the port to connect to on the remote host. ■ `scp_target` Specifies the target host and path for transferring files. Use the `user@host:/directory path/` format. This is a required parameter.
	`cat`	Concatenate files and print on the standard output. Use the `file` parameter to enter the file name.
	`grep`	Print lines that match patterns. This command requires the following parameters: ■ `pattern` Specify the naming pattern to find and select matching files and folders. For example, `spoold`, or `spad*`. This is a required parameter. ■ `file` Enter the filename. This is a required parameter. ■ `options` Enter the supported options for `grep: -i, -w, -E`, or `tail: -f, -n`.

Table 17-1 The options and arguments for the `support` command. *(continued)*

Option and its description	Argument	Description
`proc` Access process information about the pseudo-filesystem.	`tail`	Append the last 10 lines of each file to standard output. This command requires the following parameters: ■ `file` Enter the filename. This is a required parameter. ■ `options` Enter the supported options for `grep: -i, -w, -E`, or `tail: -f, -n`.
	`ls`	List information about the FILEs. Use the `dir` parameter to enter the directory name.
	`cat`	Concatenate files and print on the standard output. Use the `file` parameter to enter the file name.
	`grep`	Print lines that match patterns. This command requires the following parameters: ■ `pattern` Specify the naming pattern to find and select matching files and folders. For example, `spoold`, or `spad*`. This is a required parameter. ■ `file` Enter the filename. This is a required parameter. ■ `options` Enter the supported options for `grep: -i, -w, -E`, or `tail: -f, -n`.

Here are some examples of the support command usage:

```
support MSDP-log ls dir=spoold
support MSDP-log ls dir=spad/sof1vm08.tec.com/spoold/spad/072320.log
support MSDP-log cat file=pdde-config.log
support MSDP-log cat file=spad/sof1vm08.tec.com/spoold/spad/072320.log
support MSDP-log tail file=pdde-config.log
support MSDP-log tail file=pdde-config.log options="-f"
support MSDP-log tail file=pdde-config.log options="-n 5"
support MSDP-log tail file=spad/sofia11vm08/spoold/spad/072320.log
options="-f -n 5"
support MSDP-log grep file=spad* pattern=sessionStartAgent
options="-r"
support MSDP-log grep file=pdde-setup.log pattern="pid  455"
support MSDP-log grep file=pdde-setup.log pattern="Pid" options="-w
 -i"
support MSDP-log grep file=spoold* pattern="Pid" options="-w -i -r"
support MSDP-log grep file=pdde-setup.log pattern="MSDP-X|PureDisk"
 options="-w -i -r -E"
```

To collect logs

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 the `collect` command for the desired category to collect files of interest. For example:

   ```
   support MSDP-log collect pattern=spoold* mmin="+2"
   support MSDP-history collect
   support MSDP-config collect
   support syslogs collect pattern=crash 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.

Chapter 18

Troubleshooting

This chapter includes the following topics:

- About unified logging
- About legacy logging
- NetBackup MSDP log files
- Troubleshooting MSDP installation issues
- 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

About unified logging

Unified logging creates log file names and messages in a format that is standardized across Veritas 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

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 564.

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
TID	Long Integer	Provide the thread ID	TID = 2874950

Table 18-1 Fields in `vxlogview` query strings *(continued)*

Field name	Type	Description	Example
`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`
`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 Veritas products.

Examples of using vxlogview to view unified logs

The following examples demonstrate how to use the `vxlogview` command to view unified logs.

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`

Table 18-3 Example uses of the vxlogview command *(continued)*

Item	Example
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 Logging Assistant or the `mklogdir` batch files 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: It is recommended to always use the `mklogdir` utility present in Windows and Linux to create the legacy log directories for each platform, in order to have appropriate permissions on them.

You can use the following batch files 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.
- If any process runs for a non-root or non-admin user and there is no logging that occurs in the legacy log folders, 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 master 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:

http://www.veritas.com/docs/DOC5332

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 565.

See "About unified logging" on page 561.

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:

http://www.veritas.com/docs/DOC5332

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 Veritas representative. A log level of 5 is for troubleshooting only.

Specify the NetBackup log levels in the **Logging** host properties on the NetBackup master 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 208. See "Editing the MSDP pd.conf file" on page 209.
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 231.
		See "About recovering the MSDP catalog" on page 462.

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 Veritas 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 Veritas 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 425.
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 master 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, Veritas recommends that you set the logging level for OID 387 to 2 or less. To configure unified logs, see the following guide: The *NetBackup Logging Reference Guide*: http://www.veritas.com/docs/DOC5332

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 208. See “Editing the MSDP pd.conf file” on page 209.

Troubleshooting MSDP installation issues

The following sections may help you troubleshoot installation issues.

See “MSDP installation on SUSE Linux fails” on page 573.

MSDP installation on SUSE Linux fails

The installation trace log shows an error when you install on SUSE Linux:

```
....NetBackup and Media Manager are normally installed in /usr/openv.
Is it OK to install in /usr/openv? [y,n] (y)

Reading NetBackup files from /net/nbstore/vol/test_data/PDDE_packages/
suse/NB_FID2740_LinuxS_x86_20090713_6.6.0.27209/linuxS_x86/anb

/net/nbstore/vol/test_data/PDDE_packages/suse/NB_FID2740_LinuxS_x86_
20090713_6.6.0.27209/linuxS_x86/catalog/anb/NB.file_trans: symbol
lookup error: /net/nbstore/vol/test_data/PDDE_packages/suse/
NB_FID2740_LinuxS_x86_20090713_6.6.0.27209/linuxS_x86/catalog/anb/
NB.file_trans: undefined symbol: head /net/nbstore/vol/test_data/
PDDE_packages/suse/NB_FID2740_LinuxS_x86_20090713_6.6.0.27209/
linuxS_x86/catalog/anb/NB.file_trans failed. Aborting ...
```

Verify that your system is at patch level 2 or later, as follows:

```
cat /etc/SuSE-release
SUSE Linux Enterprise Server 10 (x86_64)
VERSION = 10
PATCHLEVEL = 2
```

Troubleshooting MSDP configuration issues

The following sections may help you troubleshoot configuration issues.

See "NetBackup MSDP log files" on page 567.

See "MSDP storage server configuration fails" on page 574.

See "MSDP database system error (220)" on page 574.

See "MSDP server not found error" on page 575.

See "License information failure during MSDP configuration" on page 575.

See "The disk pool wizard does not display an MSDP volume" on page 576.

MSDP storage server configuration fails

If storage server configuration fails, first resolve the issue that the **Storage Server Configuration Wizard** 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 229.

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 567.

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 Configuration Wizard**; 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 it does not use reverse name lookup.

To prohibit reverse host name lookup by using the Administration Console

1. In the **NetBackup Administration Console**, expand **NetBackup Management > Host Properties > Master Servers**.
2. In the details pane, select the master server.
3. On the **Actions** menu, select **Properties**.
4. In the **Master Server Properties** dialog box, select the **Network Settings** properties.
5. Select one of the following options:
   - **Allowed**
   - **Restricted**
   - **Prohibited**

   For a description of these options, see the NetBackup online Help or the administrator's guide.

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 wizard does not display an MSDP volume

The **Disk Pool Configuration Wizard** 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, restart the **NetBackup Administration Console**. 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 577.

See “MSDP backup or duplication job fails” on page 577.

See “MSDP client deduplication fails” on page 579.

See “MSDP volume state changes to DOWN when volume is unmounted” on page 580.

See “MSDP errors, delayed response, hangs” on page 581.

See “Cannot delete an MSDP disk pool” on page 581.

See “MSDP media open error (83)” on page 582.

See “MSDP media write error (84)” on page 584.

See “MSDP no images successfully processed (191)” on page 586.

See “MSDP storage full conditions” on page 586.

See “Troubleshooting MSDP catalog backup” on page 587.

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 40.

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 master 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 master 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 588.

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 446.

See "NetBackup MSDP log files" on page 567.

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 567.

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, Veritas recommends that you use fully-qualified domain names for your NetBackup environment.

See "Use fully qualified domain names" on page 54.

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 master 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 444.

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 40.

For virtual machines, Veritas 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, Veritas 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 master 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 master 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 **NetBackup Administration Console 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?

See “About MSDP performance” on page 49.

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 master 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 master 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 Networks** 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 master 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 567.

Restore or duplication fails

The `media open error (83)` message appears in the **NetBackup Administration Console 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. For a description of the problem and the procedures to resolve it, see the following Veritas tech note:

http://www.veritas.com/docs/TECH207194

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. See "Viewing MSDP disk reports" on page 417.
Data removal is running.	Data cannot be backed up at the same time as it is removed. See "About MSDP queue processing" on page 446.
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.
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 60.
The deduplication pool is down.	Change the state to up. See "Changing OpenStorage disk pool state" on page 435.
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 229.

To fix this problem, configure your network environment so that all permutations of the storage server name resolve.

Veritas recommends that you use fully qualified domain names.

See “Use fully qualified domain names” on page 54.

MSDP no images successfully processed (191)

The `no images successfully processed (191)` message appears in the **NetBackup Administration Console 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. For a description of the problem and the procedures to resolve it, see the following Veritas tech note:

http://www.veritas.com/docs/TECH207194

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 414.

See “About MSDP container files” on page 416.

See “Viewing storage usage within MSDP container files” on page 416.

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 446.

See “About the MSDP data removal process” on page 455.

Although not advised, you can reclaim free space manually.

See “Processing the MSDP transaction queue manually” on page 446.

Troubleshooting MSDP catalog backup

The following subsections provide information about MSDP catalog backup and recovery.

Catalog backup

Table 18-8 describes 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.
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 Administration Console.
- The *NetBackup Status Codes Reference Guide* available through the following webpage:
  http://www.veritas.com/docs/DOC5332

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.

See “Restoring the MSDP catalog from a shadow copy” on page 463.

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.
  See "Viewing MSDP disk reports" on page 417.
- 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 Administration Console.

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.`

Table 18-9 MSDP event codes and messages *(continued)*

Event #	Event Severity	NetBackup Severity	Message example
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.`
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.`

Table 18-9 MSDP event codes and messages *(continued)*

Event #	Event Severity	NetBackup Severity	Message example
1017	1	Critical	Service Veritas 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 Veritas Deduplication Engine completed successfully on server1.example.com.
1019	1	Critical	Service Veritas 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 Veritas 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 Veritas 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.
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.

Table 18-9 MSDP event codes and messages *(continued)*

Event #	Event Severity	NetBackup Severity	Message example
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 "Viewing MSDP disk reports" on page 417.

See "Troubleshooting MSDP operational issues" on page 576.

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 592.

See "MSDP storage server is down when you configure an OpenStorage server" on page 592.

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 245.

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
MultiDomainvconfiguration 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.

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.

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, Veritas recommends that you age the backup images on the other storage until they expire. Veritas 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 MSDP server-side deduplication" on page 69.

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*: http://www.veritas.com/docs/DOC5332
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

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 master server installations. After you upgrade the NetBackup master 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 NetBackup Administration Console or 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 "About the media server deduplication (MSDP) node cloud tier" on page 20.

See "Planning your MSDP deployment" on page 30.

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 604.

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 603.
- See "About beginning the direct migration" on page 604.
- See "Placing the Cloud Catalyst server in a consistent state" on page 605.
- See "About installing and configuring the new MSDP direct cloud tier server" on page 607.
- See "Running the migration to the new MSDP direct cloud tier server" on page 609.

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.

For more information about the minimum requirements for a new MSDP direct cloud tier storage server:

See "About the media server deduplication (MSDP) node cloud tier" on page 20.

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 272.

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 master 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 master 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 Veritas 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 Veritas Support for assistance as this issue must be corrected before continuing.

On the master 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 master 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 446.

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 Veritas 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 Veritas 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 604.

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 604.

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 30.

Note: Although not usually recommended, in some special circumstances Cloud Catalyst is running on the master server. Since you cannot uninstall, reimage the master 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 master 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 master 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.

See "About requirements for a new MSDP direct cloud tier storage server" on page 603.

See "About the media server deduplication (MSDP) node cloud tier" on page 20.

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 272.

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 604.

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 master 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 Veritas 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. Veritas 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 607.

`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.

Table B-1 Migration prompts *(continued)*

Prompts	Description
`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.
`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, NetBackup Administration Console, 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. Veritas 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 `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 master 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 master 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 master 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 master 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. Run the **Recover the catalogs** wizard in the NetBackup Administration Console.

   In the NetBackup Administration Console, click **NetBackup Management** in the left pane and then **Recover the catalogs** in the right pane. The **Catalog Recovery Wizard Welcome** panel appears.
3. Select the catalog backup image that was created before running the `nbdecommission -migrate_cloudcatalyst` command to migrate Cloud Catalyst to MSDP cloud tier server.
4. Complete all steps in the wizard to recover the NetBackup catalog.
5. Stop and restart the NetBackup services on the master server.
6. 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`.
7. Start the NetBackup services on the Cloud Catalyst server.
8. 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.
9. 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 master server:

   ```
   /usr/openv/netbackup/bin/admincmd/bpnbaz -ProvisionCert
   <CloudCatalyst host-name>
   ```

   You must restart the NetBackup services on the Cloud Catalyst server.

10 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.
```

11 (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 Run the **Recover the catalogs** wizard in the NetBackup Administration Console.

In the NetBackup Administration Console, click **NetBackup Management** in the left pane and then **Recover the catalogs** in the right pane. The **Catalog Recovery Wizard Welcome** panel appears.

3 Select the catalog backup image that was created before running the `nbdecommission -migrate_cloudcatalyst` command to migrate Cloud Catalyst to MSDP cloud tier server.

4 Complete all steps in the wizard to recover the NetBackup catalog.

5 Stop and restart the NetBackup services on the master server.

6 Reinstall the Cloud Catalyst server using the same NetBackup version and EEB bundles that were active when migration was performed.

7 Then contact Veritas 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.)

8 (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 master 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 master 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 master 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 master server catalog to revert to Cloud Catalyst.

If you do not recover the master 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 master 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 Veritas 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

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 KMS service” on page 91.

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. See "About provisioning the storage for MSDP" on page 60.

Table C-1 `crcontrol` command options *(continued)*

Option	Description
`--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 60.
`--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. ■ 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.
`--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. See "About provisioning the storage for MSDP" on page 60.

For more information about the `crcontrol`, refer to the following:

NetBackup Commands Reference Guide

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 637. See "Command usage example outputs" on page 638.

Table C-2 Encryption Crawler monitor *(continued)*

Item	Description
Data Size Scanned	The aggregated data size of the scanned data containers for **Containers Scanned**.
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. Veritas 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 633.

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 628.

See "About the two modes of the Encryption Crawler" on page 626.

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.	Veritas 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 628.

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. Veritas 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 129.

   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 626.

   See "Managing the Encryption Crawler" on page 628.

   See "Tuning options" on page 634.

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 638.

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.

Veritas 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 LinuxS or Flex WORM setups.

Table C-5

OS and Python requirements	Details
Python requirements for `encryption_reporting` on Linux 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 BYO installations.	NetBackup 10.0 and newer versions require you to install Python 3.6.8-3.9.6. Currently, no additional software packages are required to be installed. Installing Python 3.6.8-3.9.6 1 Download the Python Windows Installer for Python 3.6.8-3.9.6 from: https://www.python.org/downloads/ 2 Run the installer and select **Add Python 3.X.X to PATH**. 3 Run `python encryption_reporting.py` command by navigating to the directory containing the file (`..\Veritas\pdde`).

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.

Veritas 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
```

Index

A

Active secondary operations 198
adding trusted primary server using external certificate 176
Advanced Encryption Standard (AES) encryption 130
AES encryption
    Blowfish encryption 138
AES-256 135
appliance deduplication 17
attributes
    clearing deduplication pool 442
    clearing deduplication storage server 426
    OptimizedImage 48
    setting deduplication pool 436
    setting deduplication storage server 424
    viewing deduplication pool 435
    viewing deduplication storage server 423
Auto Image Replication
    about 158
    and trusted primary servers 165
    Backup operation in source domain 156, 158
    configuring MSDP replication to a different domain 156
    for MSDP 182
    no disk spanning support 158
    overview 158
    synchronizing clocks of master servers 159
    Targeted 179
    targeted 166
    topology of storage 162

B

Backup
    operation 197
backup
    client deduplication process 480
big endian 467, 470
bpstsinfo command 163
byte order 467, 470

C

cache hits field of the job details 413
capacity and usage reporting for deduplication 414
case sensitivity
    in NetBackup names 31
catalog, MSDP. *See* about recovering the MSDP catalog. *See* MSDP catalog
changing deduplication server hostname 427
changing the deduplication storage server name and path 427
CIFS 35
clearing deduplication pool attributes 442
client deduplication
    components 479
    disabling for a specific client 127
    host requirements 44
clustering
    primary server inter-node authentication 179
Common Internet File System 35
compacting container files 416
compression
    for MSDP backups 128
    for MSDP optimized duplication and replication 128
    pd.conf file setting 210
configuring a deduplication pool 114
configuring a deduplication storage server 95
configuring a deduplication storage unit 121
configuring deduplication 69, 71
container files
    about 416
    compaction 416
    viewing capacity within 416
contentrouter.cfg file
    about 224
    parameters for data integrity checking 452
    ServerOptions for encryption 133
Coordinated Universal Time (UTC) 159
CR sent field of the job details 413–414
crcontrol 136

credentials 46
adding NetBackup Deduplication Engine 432
changing NetBackup Deduplication Engine 433

D

data classifications
in storage lifecycle policies 190, 197
data conversion
encryption 135
data integrity checking
about deduplication 448
configuration settings for deduplication 451
configuring behavior for deduplication 448
data removal process
for deduplication 455
database system error 574
deactivating media server deduplication 473
dedup field of the job details 414
deduplication
about credentials 46
about fingerprinting 80
about the license 64
adding credentials 432
and Fibre Channel 36
and iSCSI 36
cache hits field of the job details 413
capacity and usage reporting 414
changing credentials 433
client backup process 480
configuration file 208
configuring 69, 71
configuring optimized synthetic backups 139
container files 416
CR sent field of the job details 413–414
data removal process 455
dedup field of the job details 414
event codes 588
license for 64
licensing 65
limitations 42
media server process 478
network interface 47
node 32
performance 49
planning deployment 30
requirements for optimized within the same domain 142
scaling 55
scanned field of the job details 414
deduplication *(continued)*
storage capacity 33
storage destination 32
storage management 62
storage requirements 34
storage unit properties 122
deduplication configuration file
editing 79, 209
parameters 210
deduplication data integrity checking
about 448
configuring behavior for 448
settings 451
deduplication deduplication pool. *See* deduplication pool
deduplication disk volume
changing the state 444
determining the state 443
deduplication encryption
enabling for MSDP backups 132
deduplication host configuration file 229
deleting 229
deduplication hosts
and firewalls 47
client requirements 44
load balancing server 40
server requirements 40
storage server 39
deduplication installation
log file 571
deduplication logs
about 567
client deduplication proxy plug-in log 569
client deduplication proxy server log 569
configuration script 569
deduplication plug-in log 570
NetBackup Deduplication Engine 571
NetBackup Deduplication Manager 572
VxUL deduplication logs 567
deduplication node
about 32
adding a load balancing server 205
removing a load balancing server 429
deduplication optimized synthetic backups
about 48
deduplication plug-in
about 476
log file 570

deduplication plug-in configuration file
configuring 74
deduplication pool. *See* deduplication pool
about 113
changing properties 437
clearing attributes 442
configuring 114
deleting 445
determining the state 434
properties 116
setting attributes 436
viewing 434
viewing attributes 435
deduplication port usage
about 47
troubleshooting 585
deduplication processes do not start 577
deduplication rate
how file size affects 50
deduplication reference database
about 476
deduplication registry
resetting 230
deduplication servers
components 475
host requirements 40
deduplication storage capacity
about 33
viewing capacity in container files 416
deduplication storage destination 32
deduplication storage requirements 34
deduplication storage server
about 39
change the name 427
changing properties 425
clearing attributes 426
components 475
configuration failure 575
configuring 95
defining target for Auto Image Replication 162
deleting 430
deleting the configuration 431
determining the state 422
editing configuration file 227
getting the configuration 226
recovery 466
replacing the host 470
setting attributes 424
setting the configuration 228
deduplication storage server *(continued)*
viewing 422
viewing attributes 423
deduplication storage server configuration file
about 225
deduplication storage server name
changing 427
deduplication storage type 32
Deduplication storage unit
Only use the following media servers 123
Use any available media server 123
deduplication, compression rate
monitoring 409
deleting backup images 445
deleting deduplication host configuration file 229
df command 586
disaster recovery
protecting the data 59
recovering the storage server after catalog recovery 469
disk failure
deduplication storage server 465
disk logs 418
disk logs report 416
disk pool status report 415, 418
disk pools
cannot delete 581
disk storage unit report 418
Disk type 122
disk volume
changing the state 444
determining the state of a deduplication 443
volume state changes to down 580
domains
replicating backups to another. *See* Auto Image Replication
Duplication
optimized 198
duplication jobs, cancelling 582
Duplication to remote master. *See* Auto Image Replication

E

Enable file recovery from VM backup 199
encryption
enabling for MSDP backups 132
pd.conf file setting 213
SHA-2 80, 135

endian
big 467, 470
little 467, 470
event codes
deduplication 588

F

Fibre Channel
and iSCSI comparison 36
support for 35
Fibre Channel and iSCSI comparison for MSDP 36
file system
CIFS 35
NFS 35
Veritas File System for deduplication storage 62
ZFS 35
fingerprinting
about deduplication 80
firewalls and deduplication hosts 47
FlashBackup policy
Maximum fragment size (storage unit setting) 123
FQDN or IP Address property in Resilient Network host properties 201

G

garbage collection. *See* queue processing

H

host requirements 40

I

images on disk report 418
Import
operation 191
iSCSI
and Fibre Channel comparison 36
support for 35

J

job ID search in unified logs 565

L

legacy logging 565
license
for deduplication 64
license information failure
for deduplication 575
licensing deduplication 65
limitations
media server deduplication 42
little endian 467, 470
load balancing server
about 40
adding to a deduplication node 205
for deduplication 40
removing from deduplication node 429
logging
legacy 565
logs
about deduplication 567
Auto Image Replication 572
client deduplication proxy plug-in log 569
client deduplication proxy server log 569
deduplication configuration script log 569
deduplication installation 571
deduplication plug-in log 570
disk 418
NetBackup Deduplication Engine log 571
NetBackup Deduplication Manager log 572
optimized duplication 572
VxUL deduplication logs 567

M

maintenance processing. *See* queue processing
Maximum concurrent jobs 124
Maximum fragment size 123
media server deduplication
process 478
Media Server Deduplication Pool 113, 182
creating directories for 400 TB support 118
enable 400 TB support 90
media server deduplication pool. *See* deduplication pool
migrating to NetBackup deduplication 595
mklogdir.bat 565
MSDP
replication 155
replication target, configuring 182
MSDP catalog 231, 462
See also MSDP catalog backup
See also MSDP catalog recovery
about the catalog backup policy 232
about the shadow catalog 231
changing the number of shadow copies 236
changing the shadow catalog path 234
changing the shadow catalog schedule 235

MSDP catalog *(continued)*
shadow copy log files 568
MSDP catalog backup
about protecting the MSDP catalog 232
configuring 237, 242
MSDP catalog recovery
about 462
error codes 587
process the transation queue. 447
recover from a shadow copy 463
MSDP drcontrol utility
options 238
MSDP replication
about 49
MSDP storage rebasing. *See* rebasing
mtstrm.conf file
configuring 74

N

nbstserv process 192
NDMP
storage units 198
NetBackup
naming conventions 31
NetBackup 5200 series appliance
as a storage destination 33
NetBackup 5300 series appliance
as a storage destination 33
NetBackup appliance deduplication 17
NetBackup deduplication
about 16
license for 64
NetBackup Deduplication Engine
about 476
about credentials 46
adding credentials 432
changing credentials 433
logs 571
NetBackup Deduplication Manager
about 476
logs 572
NetBackup deduplication options 16
network interface
for deduplication 47
NFS 35
node
deduplication 32

O

OpenStorage
optimized duplication 198
OpenStorage Disk Option 190
optimized deduplication
configuring bandwidth 188
configuring for MSDP 148
limitations 142
logs 572
pull configuration within the same domain 146
separate network for 140
optimized deduplication copy
guidance for 144
optimized duplication
about 49
about the media server in common within the same domain 143
push configuration within the same domain 144
optimized duplication encryption
configuring for MSDP 134
optimized MSDP deduplication
requirements 142
within the same domain 141
optimized synthetic backups
configuring for deduplication 139
deduplication 48
OptimizedImage attribute 48

P

pd.conf file
about 208
editing 79, 209
parameters 210
pdde-config.log 569
performance
deduplication 49
monitoring deduplication rate 409
port usage
and deduplication 47
troubleshooting 585
Postponed secondary operations 198
primary servers
inter-node authentication for clustering 179
Priority for secondary operations 197
provisioning the deduplication storage 60
PureDisk Deduplication Pool 113

Q

queue processing 446
    invoke manually 447

R

rebasing
    about 453
    FP_CACHE_PERIOD_REBASING_THRESHOLD parameter 216
    FP_CACHE_REBASING_THRESHOLD parameter 216
    RebaseMaxPercentage parameter 455
    RebaseMaxTime parameter 455
    RebaseMinContainers parameter 455
    RebaseScatterThreshold parameter 455
    server-side rebasing parameters 455
recovery
    deduplication storage server 466
    from deduplication storage server disk failure 465
Red Hat Linux
    deduplication processes do not start 577
replacing the deduplication storage server 470
replication
    between NetBackup domains. *See* Auto Image Replication
    configuring MSDP replication to a different domain 156
    for MSDP 49, 155
    to an alternate NetBackup domain. *See* Auto Image Replication
replication encryption
    configuring for MSDP 134
reports
    disk logs 416, 418
    disk pool status 415, 418
    disk storage unit 418
resetting the deduplication registry 230
Resiliency property in Resilient Network host properties 201
Resilient connection
    Resilient Network host properties 199
resilient network connection
    log file 572
Resilient Network host properties 199
    FQDN or IP Address property in 201
    Resiliency property in 201
restores
    at a remote site 459
    how deduplication restores work 457
restores *(continued)*
    specifying the restore server 460
reverse host name lookup
    prohibiting 575
reverse name lookup 575
rolling conversion
    AES encryption 135

S

scaling deduplication 55
scanned field of the job details 414
Secure Hash Algorithm 80, 135
server not found error 575
setting deduplication pool attributes 436
SHA-2 80, 135
SHA-512/256 135
SLP Parameters 154
snapshots
    operation type 197
spa.cfg file
    parameters for data integrity checking 452
storage capacity
    about 33
    for deduplicaton 33
    viewing capacity in container files 416
Storage Lifecycle Manager service (nbstserv) 192
storage lifecycle policies
    Active secondary operations 198
    best practices document 190
    cancelling duplication jobs 582
    Data classification setting 197
    hierarchy 194
    operations 192
    Postponed secondary operations 198
    Priority for secondary operations 197
    Storage lifecycle policy name 196
    utility 190
    Validate Across Backup Policies button 198
storage paths
    about reconfiguring 427
    changing 427
storage rebasing. *See* rebasing
storage requirements
    for deduplication 34
storage server
    about the configuration file 225
    change the name 427
    changing properties for deduplication 425
    changing the name 427

storage server *(continued)*
components for deduplication 475
configuring for deduplication 95
deduplication 39
define target for Auto Image Replication 162
deleting a deduplication 430
deleting the deduplication configuration 431
determining the state of a deduplication 422
editing deduplication configuration file 227
getting deduplication configuration 226
recovery 466
replacing the deduplication host 470
setting the deduplication configuration 228
viewing 422
viewing attributes 423
storage server configuration
getting 226
setting 228
storage server configuration file
editing 227
storage type
for deduplication 32
storage unit
configuring for deduplication 121
properties for deduplication 122
recommendations for deduplication 124
storage unit groups
and storage lifecycle polices 198
not supported for Auto Image Replication source 156, 158
Storage unit name 122
Storage unit type 122
stream handlers
NetBackup 50
synthetic backups
no Auto Image Replication support 158

T

Targeted A.I.R. 179
topology of storage 162–163
troubleshooting
database system error 574
deduplication backup jobs fail 577
deduplication processes do not start 577
general operational problems 581
host name lookup 575
installation fails on Linux 573
no volume appears in disk pool wizard 576
server not found error 575
trusted master servers
adding 171
trusted primary servers
for Auto Image Replication 165
removing 177

U

unified logging 561
format of files 562
uninstalling media server deduplication 473
UTC (Coordinated Universal Time) 159

V

Validate Across Backup Policies button in SLP 198
Validation Report tab 198
viewing deduplication pool attributes 435
viewing storage server attributes 423
VM backup 199
volume manager
Veritas Volume Manager for deduplication storage 62
vxlogview command 562
with job ID option 565

Z

ZFS 35