ONTAP® 9

NFS Configuration Power Guide

October 2016 | [215-11588_A0]
doccomments@netapp.com

Updated for ONTAP 9.1
Release Candidate Documentation - Contents Subject To Change

Contents

Deciding whether to use the NFS Configuration Power Guide

This guide describes how to use ONTAP 9.0 CLI commands to configure NFS client access to files contained in a new volume or qtree in a new or existing SVM. It includes examples and advanced configuration options.

You should use this guide if you want to configure access to a volume or qtree in the following way:

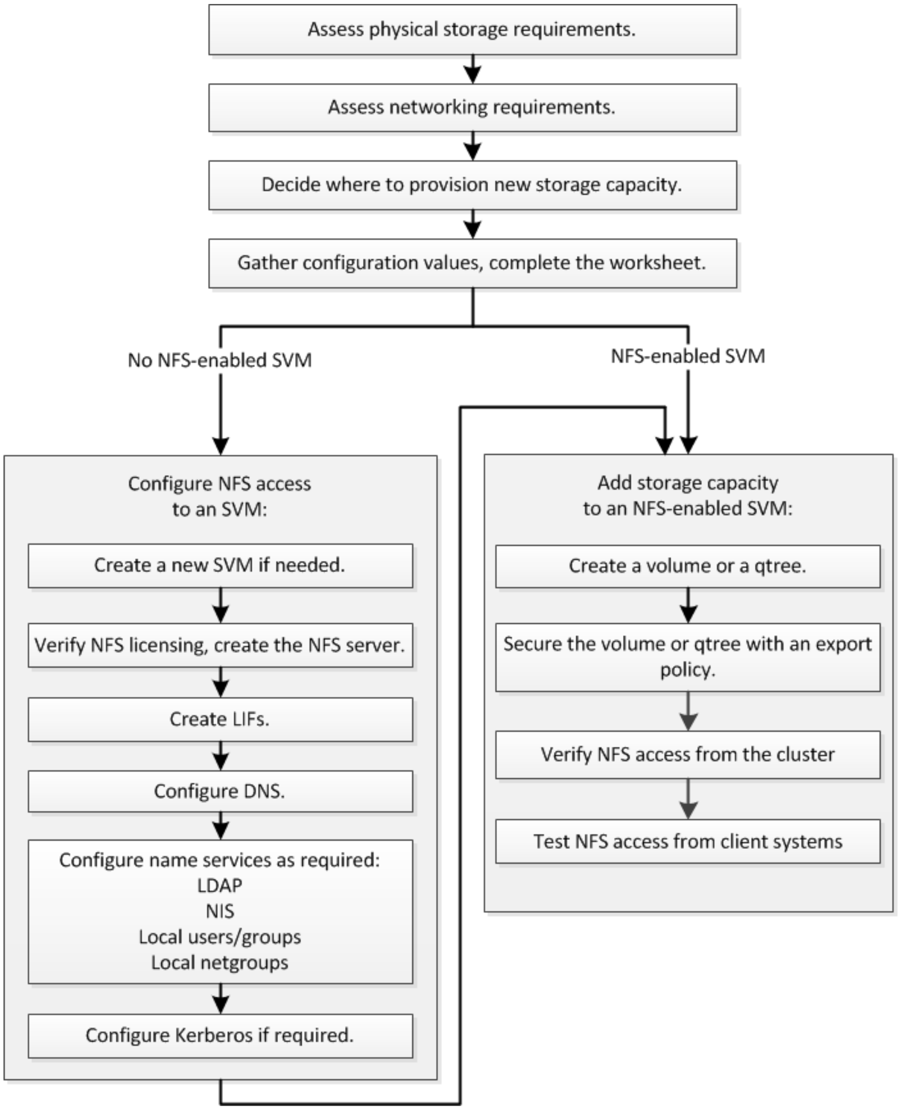
- You want to use any version of NFS currently supported by ONTAP: NFSv3, NFSv4, NFSv4.1, or NFSv4.1 with pNFS.
- You want to use the command-line interface (CLI), not OnCommand System Manager or an automated scripting tool.
  You can use the *NFS Configuration Express Guide* and other Express Guides to support configuration with System Manager, and OnCommand Workflow Automation for automated scripting support.
- You want to use best practices, not explore every available option.
  Details about command syntax are available from CLI help and Data ONTAP man pages.
- You do not want to read a lot of conceptual background.
  Additional information about Data ONTAP technology and interaction with external services is available in the Data ONTAP Reference Library and in Technical Reports (TRs).
- UNIX file permissions will be used to secure the new volume.
- You want to provision storage on a FlexVol volume or a qtree, not an Infinite Volume.
- You have cluster administrator privileges, not SVM administrator privileges.

If this guide is not suitable for your situation, you should see the following documentation instead:

- *NFS express configuration*
- *ONTAP 9 Commands: Manual Page Reference*
- *NFS management*
- *ONTAP 9 Network Management Guide*
- *NetApp Technical Report 4067: Clustered Data ONTAP Best Practice and NFS Implementation Guide*
- *NetApp Technical Report 4073: Secure Unified Authentication with NetApp Storage Systems: Kerberos, NFSv4, and LDAP for User Authentication over NFS (with a Focus on Clustered Data ONTAP)*
- *NetApp Technical Report 3580: NFSv4 Enhancements and Best Practices Guide: Data ONTAP Implementation*
- *NetApp Technical Report 4379: Name Services Best Practice Guide Clustered Data ONTAP*
- *NetApp Documentation: OnCommand Workflow Automation (current releases)*
  OnCommand Workflow Automation enables you to run prepackaged workflows that automate management tasks such as the workflows described in Express Guides.

NFS configuration workflow

Configuring NFS involves assessing physical storage and networking requirements, and then choosing a workflow that is specific to your goal—configuring NFS access to a new or existing SVM, or adding a volume or qtree to an existing SVM that is already fully configured for NFS access.


Assess physical storage requirements.
Assess networking requirements.
Decide where to provision new storage capacity.
Gather configuration values, complete the worksheet.
No NFS-enabled SVM
NFS-enabled SVM
Configure NFS access to an SVM:
Create a new SVM if needed.
Verify NFS licensing, create the NFS server.
Create LIFs.
Configure DNS.
Configure name services as required: LDAP NIS Local users/groups Local netgroups
Configure Kerberos if required.
Add storage capacity to an NFS-enabled SVM:
Create a volume or a qtree.
Secure the volume or qtree with an export policy.
Verify NFS access from the cluster
Test NFS access from client systems

Assessing physical storage requirements

Before provisioning NFS storage for clients, you must ensure that there is sufficient space in an existing aggregate for the new volume. If there is not, you can add disks to an existing aggregate or create a new aggregate.

Steps

1. Display available space in existing aggregates:

`storage aggregate show`

If there is an aggregate with sufficient space, record its name in the worksheet.

Example

```
cluster::> storage aggregate show
Aggregate     Size Available Used% State   #Vols  Nodes  RAID Status
--------- -------- --------- ----- ------- ------ ------ -----------
aggr_0     239.0GB   11.13GB   95% online       1 node1  raid_dp,
                                                         normal
aggr_1     239.0GB   11.13GB   95% online       1 node1  raid_dp,
                                                         normal
aggr_2     239.0GB   11.13GB   95% online       1 node2  raid_dp,
                                                         normal
aggr_3     239.0GB   11.13GB   95% online       1 node2  raid_dp,
                                                         normal
aggr_4     239.0GB   238.9GB   95% online       5 node3  raid_dp,
                                                         normal
aggr_5     239.0GB   239.0GB   95% online       4 node4  raid_dp,
                                                         normal
6 entries were displayed.
```

2. If there are no aggregates with sufficient space, add disks to an existing aggregate by using the `storage aggregate add-disks` command or create a new one by using the `storage aggregate create` command.

Related information

ONTAP 9 man page: storage aggregate create
ONTAP 9 Disks and Aggregates Power Guide

Assessing networking requirements

Before providing NFS storage to clients, you must verify that networking is correctly configured to meet the NFS provisioning requirements.

Before you begin

The following cluster networking objects must be configured:

- Physical and logical ports
- Broadcast domains
- Subnets (if required)
- IPspaces (as required, in addition to the default IPspace)
- Failover groups (as required, in addition to the default failover group)
- External firewalls

Steps

1. Display the available physical and virtual ports:

   `network port show`

   - When possible, you should use the port with the highest MTU setting.
   - If you are using virtual ports, you should verify that the MTU of the virtual port matches that of the underlying physical ports.

2. If you are planning to use a subnet name to allocate the IP address and network mask value for a LIF, verify that the subnet exists and has sufficient addresses available:

   ```
   network subnet show
   ```

   Subnets contain a pool of IP addresses that belong to the same layer 3 subnet. Subnets are created by using the `network subnet create` command.

3. Display available IPspaces:

   ```
   network ipspace show
   ```

   You can use the default IPspace or a custom IPspace.

4. If you want to use IPv6 addresses, verify that IPv6 is enabled on the cluster:

   ```
   network options ipv6 show
   ```

   If required, you can enable IPv6 by using the `network options ipv6 modify` command.

Related information

ONTAP 9 man page: network port show
ONTAP 9 man page: network subnet show
ONTAP 9 man page: network ipspace show
ONTAP 9 man page: network options ipv6 modify

Deciding where to provision new NFS storage capacity

Before you create a new NFS volume or qtree, you must decide whether to place it in a new or existing SVM, and how much configuration the SVM requires. This decision determines your workflow.

Choices

- If you want to provision a volume or qtree on a new SVM, or on an existing SVM that has NFS enabled but not configured, complete the steps in both “Configuring NFS access to an SVM” and “Adding NFS storage to an NFS-enabled SVM”.

  Configuring NFS access to an SVM on page 17

  Adding NFS storage to an NFS-enabled SVM on page 45

  You might choose to create a new SVM if one of the following is true:

  - You are enabling NFS on a cluster for the first time.
  - You have existing SVMs in a cluster in which you do not want to enable NFS support.
  - You have one or more NFS-enabled SVMs in a cluster, and you want another NFS server in an isolated namespace (multi-tenancy scenario).

  You should also choose this option to provision storage on an existing SVM that has NFS enabled but not configured. This might be the case if you created the SVM for SAN access or if no protocols were enabled when the SVM was created.

  After enabling NFS on the SVM, proceed to provision a volume or qtree.

- If you want to provision a volume or qtree on an existing SVM that is fully configured for NFS access, complete the steps in “Adding NFS storage to an NFS-enabled SVM”.

  Adding NFS storage to an NFS-enabled SVM on page 45

Worksheet for gathering NFS configuration information

The NFS configuration worksheet enables you to collect the required information to set up NFS access for clients.

You should complete one or both sections of the worksheet depending on the decision you made about where to provision storage:

- If you are configuring NFS access to an SVM, you should complete both sections.
  Configuring NFS access to an SVM on page 9
  Adding storage capacity to an NFS-enabled SVM on page 14
- If you are adding storage capacity to an NFS-enabled SVM, you should complete only the second section.

Adding storage capacity to an NFS-enabled SVM on page 14

See the command man pages for details about the parameters.

Configuring NFS access to an SVM

Parameters for creating an SVM

You supply these values with the `vserver create` command if you are creating a new SVM.

Field	Description	Your value
`-vserver`	A name you supply for the new SVM that is either a fully qualified domain name (FQDN) or follows another convention that enforces unique SVM names across a cluster.	
`-aggregate`	The name of an aggregate in the cluster with sufficient space for new NFS storage capacity.	
`-rootvolume`	A unique name you supply for the SVM root volume.	
`-rootvolume-security-style`	Use the UNIX security style for the SVM.	**`unix`**
`-language`	Use the default language setting in this workflow.	**`C.UTF-8`**
`ipspace`	IPspaces are distinct IP address spaces in which (Storage Virtual Machines (SVMs)) reside.	

Parameters for creating an NFS server

You supply these values with the `vserver nfs create` command when you create a new NFS server and specify supported NFS versions.

If you are enabling NFSv4 or later, you should use LDAP for improved security.

Field	Description	Your value
`-v3`, `-v4.0`, `-v4.1`, `-v4.1-pnfs`	Enable NFS versions as needed.	
`-v4-id-domain`	ID mapping domain name.	
`-v4-numeric-ids`	Support for numeric owner IDs (enabled or disabled).	

Parameters for creating a LIF

You supply these values with the `network interface create` command when you are creating LIFs.

If you are using Kerberos, you should enable Kerberos on multiple LIFs.

Field	Description	Your value
`-lif`	A name you supply for the new LIF.	
`-role`	Use the data LIF role in this workflow.	**`data`**
`-data-protocol`	Use only the NFS protocol in this workflow.	**`nfs`**
`-home-node`	The node to which the LIF returns when the `network interface revert` command is run on the LIF.	
`-home-port`	The port or interface group to which the LIF returns when the `network interface revert` command is run on the LIF.	
`-address`	The IPv4 or IPv6 address on the cluster that will be used for data access by the new LIF.	
`-netmask`	The network mask and gateway for the LIF.	
`-subnet`	A pool of IP addresses. Used instead of `-address` and `-netmask` to assign addresses and netmasks automatically.	
`-firewall-policy`	Use the default data firewall policy in this workflow.	**`data`**

Parameters for DNS host name resolution

You supply these values with the `vserver services name-service dns create` command when you are configuring DNS.

Field	Description	Your value
`-domains`	Up to five DNS domain names.	
`-name-servers`	Up to three IP addresses for each DNS name server.	

Name service information

Parameters for creating local users

You supply these values if you are creating local users by using the `vserver services name-service unix-user create` command. If you are configuring local users by loading a file containing UNIX users from a uniform resource identifier (URI), you do not need to specify these values manually.

	User name (`-user`)	**User ID (`-id`)**	**Group ID (`-primary-gid`)**	**Full name (`-full-name`)**
Example	johnm	123	100	John Miller
1				
2				
3				
...				
n				

Parameters for creating local groups

You supply these values if you are creating local groups by using the `vserver services name-service unix-group create` command. If you are configuring local groups by loading a file containing UNIX groups from a URI, you do not need to specify these values manually.

	Group name (`-name`)	**Group ID (`-id`)**
Example	Engineering	100
1		
2		
3		
...		
n		

Parameters for NIS

You supply these values with the `vserver services name-service nis-domain create` command.

Field	**Description**	**Your value**
`-domain`	The NIS domain that the SVM will use for name lookups.	
`-active`	The active NIS domain server.	**`true`** or **`false`**
`-servers`	One or more IP addresses of NIS servers used by the NIS domain configuration.	

Parameters for LDAP

You supply these values with the `vserver services name-service ldap client create` command.

You will also need a self-signed root CA certificate `.pem` file.

Field	Description	Your value
`-vserver`	The name of the SVM for which you want to create an LDAP client configuration.	
`-client-config`	The name you assign for the new LDAP client configuration.	
`-servers`	One or more LDAP servers by IP address in a comma-delimited list.	
`-query-timeout`	Use the default **`3`** seconds for this workflow.	**`3`**
`-min-bind-level`	The minimum bind authentication level. The default is **`anonymous`**. Must be set to **`sasl`** if signing and sealing is configured.	
`-preferred-ad-servers`	One or more preferred Active Directory servers by IP address in a comma-delimited list.	
`-ad-domain`	The Active Directory domain.	
`-schema`	The schema template to use. You can use a default or custom schema.	
`-port`	Use the default LDAP server port **`389`** for this workflow.	**`389`**
`-bind-dn`	The Bind user distinguished name.	
`-base-dn`	The base distinguished name. The default is **`""`** (root).	
`-base-scope`	Use the default base search scope **`subnet`** for this workflow.	**`subnet`**
`-session-security`	Enables LDAP signing or signing and sealing. The default is **none**.	
`-use-start-tls`	Enables LDAP over TLS. The default is **`false`**.	

Parameters for Kerberos authentication

You supply these values with the `vserver nfs kerberos realm create` command. Some of the values will differ depending on whether you use Microsoft Active Directory as a Key Distribution Center (KDC) server, or MIT or other UNIX KDC server.

Field	Description	Your value
`-vserver`	The SVM that will communicate with the KDC.	
`-realm`	The Kerberos realm.	
`-clock-skew`	Permitted clock skew between clients and servers.	

Field	**Description**	**Your value**
`-kdc-ip`	KDC IP address.	
`-kdc-port`	KDC port number.	
`-adserver-name`	Microsoft KDC only: AD server name.	
`-adserver-ip`	Microsoft KDC only: AD server IP address.	
`-adminserver-ip`	UNIX KDC only: Admin server IP address.	
`-adminserver-port`	UNIX KDC only: Admin server port number.	
`-passwordserver-ip`	UNIX KDC only: Password server IP address.	
`-passwordserver-port`	UNIX KDC only: Password server port.	
`-kdc-vendor`	KDC vendor.	{ **`Microsoft`** | **`Other`** }
`-comment`	Any desired comments.	

You supply these values with the `vserver nfs kerberos interface enable` command.

Field	**Description**	**Your value**
`-vserver`	The name of the SVM for which you want to create an Kerberos configuration.	
`-lif`	The data LIF on which you will enable Kerberos. You can enable Kerberos on multiple LIFs.	
`-spn`	The Service Principle Name (SPN)	
`-permitted-enc-types`	The permitted encryption types for Kerberos over NFS; **`aes-256`** is recommended, depending on client capabilities.	
`-admin-username`	The KDC administrator credentials to retrieve the SPN secret key directly from the KDC. A password is required	
`-keytab-uri`	The keytab file from the KDC containing the SPN key if you do not have KDC administrator credentials.	

Field	Description	Your value
`-ou`	The organizational unit (OU) under which the Microsoft Active Directory server account will be created when you enable Kerberos using a realm for Microsoft KDC.	

Adding storage capacity to an NFS-enabled SVM

Parameters for creating export policies and rules

You supply these values with the `vserver export-policy create` command.

Field	Description	Your value
`-vserver`	The name of the SVM that will host the new volume.	
`-policyname`	A name you supply for a new export policy.	

You supply these values for each rule with the `vserver export-policy rule create` command.

Field	Description	Your value
`-clientmatch`	Client match specification.	
`-ruleindex`	Position of export rule in the list of rules.	
`-protocol`	Use NFS in this workflow.	**`nfs`**
`-rorule`	Authentication method for read-only access.	
`-rwrule`	Authentication method for read-write access.	
`-superuser`	Authentication method for superuser access.	
`-anon`	User ID to which anonymous users are mapped.	

You must create one or more rules for each export policy.

`-ruleindex`	**`-clientmatch`**	**`-rorule`**	**`-rwrule`**	**`-superuser`**	**`-anon`**
Examples	0.0.0.0/0,@rootaccess_netgroup	any	krb5	sys	65534
1					
2					
3					
...					
n					

Parameters for creating a volume

You supply these values with the `volume create` command if you are creating a volume instead of a qtree.

Field	Description	Your value
`-vserver`	The name of a new or existing SVM that will host the new volume.	
`-volume`	A unique descriptive name you supply for the new volume.	
`-aggregate`	The name of an aggregate in the cluster with sufficient space for the new NFS volume.	
`-size`	An integer you supply for the size of the new volume.	
`-user`	Name or ID of the user that is set as the owner of the volume's root.	
`-group`	Name or ID of the group that is set as the owner of the volume's root.	
`--security-style`	Use the UNIX security style for this workflow.	**`unix`**
`-junction-path`	Location under root (/) where the new volume is to be mounted.	
`-export-policy`	If you are planning to use an existing export policy, you can enter its name when you create the volume.	

Parameters for creating a qtree

You supply these values with the `volume qtree create` command if you are creating a qtree instead of a volume.

Field	Description	Your value
`-vserver`	The name of the SVM on which the volume containing the qtree resides.	
`-volume`	The name of the volume that will contain the new qtree.	
`-qtree`	A unique descriptive name you supply for the new qtree, 64 characters or less.	
`-qtree-path`	The qtree path argument in the format `/vol/`*`volume_name/ qtree_name`*`>` can be specified instead of specifying volume and qtree as separate arguments.	
`-unix-permissions`	Optional: The UNIX permissions for the qtree.	

Field	**Description**	**Your value**
`-export-policy`	If you are planning to use an existing export policy, you can enter its name when you create the qtree.	

Configuring NFS access to an SVM

If you do not already have an SVM configured for NFS client access, you must either create and configure a new SVM or configure an existing SVM. Configuring NFS involves opening SVM root volume access, creating an NFS server, creating a LIF, enabling host-name resolution, configuring name services, and if desired, enabling Kerberos security.

Steps

1. Creating an SVM on page 17
2. Verifying that NFS is enabled on the SVM on page 18
3. Opening the export policy of the SVM root volume on page 19
4. Creating an NFS server on page 20
5. Creating a LIF on page 22
6. Enabling DNS for host-name resolution on page 24
7. Configuring name services on page 25
8. Using Kerberos with NFS for strong security on page 39

Creating an SVM

If you do not already have at least one SVM in a cluster to provide data access to NFS clients, you must create one.

Steps

1. Create an SVM:

   ```
   vserver create -vserver vserver_name -rootvolume root_volume_name -
   aggregate aggregate_name -rootvolume-security-style unix -language
   C.UTF-8 -ipspace ipspace_name
   ```

   - Use the UNIX setting for the `-rootvolume-security-style` option.
   - Use the default C.UTF-8 `-language` option.
   - The `ipspace` setting is optional.

2. Verify the configuration and status of the newly created SVM:

   ```
   vserver show -vserver vserver_name
   ```

   The `Allowed Protocols` field must include NFS. You can edit this list later.

   The `Vserver Operational State` field must display the `running` state. If it displays the `initializing` state, it means that some intermediate operation such as root volume creation failed, and you must delete the SVM and re-create it.

Examples

The following command creates an SVM for data access in the IPspace ipspaceA:

```
cluster1::>vserver create -vserver vs0.example.com -rootvolume
root_vs0 -aggregate aggr1
-rootvolume-security-style unix -language C.UTF-8 -ipspace ipspaceA

[Job 2059] Job succeeded:
Vserver creation completed
```

The following command shows that an SVM was created with a root volume of 1 GB, and it was started automatically and is in `running` state. The root volume has a default export policy that does not include any rules, so the root volume is not exported upon creation.

```
cluster1::> vserver show -vserver vs0.example.com
                                    Vserver: vs0.example.com
                               Vserver Type: data
                            Vserver Subtype: default
                               Vserver UUID: b8375669-19b0-11e5-
b9d1-00a0983d9736
                                Root Volume: root_vs0
                                  Aggregate: aggr1
                                 NIS Domain: -
                 Root Volume Security Style: unix
                                LDAP Client: -
               Default Volume Language Code: C.UTF-8
                            Snapshot Policy: default
                                    Comment:
                               Quota Policy: default
                List of Aggregates Assigned: -
 Limit on Maximum Number of Volumes allowed: unlimited
                        Vserver Admin State: running
                  Vserver Operational State: running
   Vserver Operational State Stopped Reason: -
                          Allowed Protocols: nfs, cifs, fcp, iscsi,
ndmp
                       Disallowed Protocols: -
            Is Vserver with Infinite Volume: false
                           QoS Policy Group: -
                                Config Lock: false
                               IPspace Name: ipspaceA
```

Related information

ONTAP 9 man page: vserver create

Verifying that NFS is enabled on the SVM

Before you can configure and use NFS on SVMs, you must enable the protocol. This is typically done during SVM setup, but if you did not enable the protocol during setup, you can enable it later by using the `vserver add-protocols` command.

About this task

You can also disable protocols on SVMs using the `vserver remove-protocols` command.

Steps

1. Check which protocols are currently enabled and disabled for the SVM:

   `vserver show -vserver` ***`vserver_name`*** **`-protocols`**

   You can also use the `vserver show-protocols` command to view the currently enabled protocols on all SVMs in the cluster.

2. Perform one or both of the following actions:

If you want to...	Enter the command...
Enable NFS	**vserver add-protocols -vserver *vserver_name* -protocols nfs**
Disable a protocol	**vserver remove-protocols -vserver *vserver_name* -protocols *protocol_name*[,*protocol_name*,...]**

3. Confirm that the allowed and disallowed protocols were updated correctly:

   vserver show -vserver *vserver_name* -protocols

Example

The following command displays which protocols are currently enabled and disabled on the SVM named vs1:

```
vs1::> vserver show -vserver vs1 -protocols
Vserver       Allowed Protocols          Disallowed Protocols
-----------   ----------------------     -----------------------
vs1           nfs                        cifs, fcp, iscsi, ndmp
```

The following command allows access over NFS by adding **nfs** to the list of enabled protocols on the SVM named vs1:

```
vs1::> vserver add-protocols -vserver vs1 -protocols nfs
```

Related information

ONTAP 9 man page: vserver add-protocols

Opening the export policy of the SVM root volume

The default export policy of the SVM root volume must include a rule to allow all clients access through NFS. Without such a rule, all NFS clients are denied access to the SVM and its volumes.

About this task

When a new SVM is created, a default export policy (called default) is created automatically for the root volume of the SVM. You must create one or more rules for the default export policy before clients can access data on the SVM.

You should verify that all NFS access is open in the default export policy, and later restrict access to individual volumes by creating custom export policies for individual volumes or qtrees.

Steps

1. If you are using an existing SVM, check the default root volume export policy:

   vserver export-policy rule show

   Example

   The command output should be similar to the following:

```
cluster::> vserver export-policy rule show -vserver vs0.example.com -
policyname default -instance

                                    Vserver: vs0.example.com
                                Policy Name: default
                                 Rule Index: 1
                            Access Protocol: nfs
Client Match Hostname, IP Address, Netgroup, or Domain: 0.0.0.0/0
                             RO Access Rule: any
                             RW Access Rule: any
User ID To Which Anonymous Users Are Mapped: 65534
                   Superuser Security Types: any
               Honor SetUID Bits in SETATTR: true
                  Allow Creation of Devices: true
```

If such a rule exists that allows open access, this task is complete. If not, proceed to the next step.

2. Create an export rule for the SVM root volume:

   vserver export-policy rule create -vserver *vserver_name* -policyname default -ruleindex 1 -ruleindex 1 -protocol nfs -clientmatch 0.0.0.0/0 -rorule any -rwrule any -superuser any

   If the SVM will only contain volumes secured by Kerberos, you can set the export rule options **-rorule**, **-rwrule**, and **-superuser** for the root volume to **krb5** or **krb5i**. For example:

   -rorule krb5i -rwrule krb5i -superuser krb5i

3. Verify rule creation by using the `vserver export-policy rule show` command.

Result

Any NFS client can now access any volume or qtree created on the SVM.

Related information

ONTAP 9 man page: vserver export-policy rule create

Creating an NFS server

After verifying that NFS is licensed on your cluster, you can use the `vserver nfs create` command to create an NFS server on the SVM and specify the NFS versions it supports.

Before you begin

The SVM must have been configured to allow the NFS protocol.

About this task

The SVM can be configured to support one or more versions of NFS. If you are supporting NFSv4 or later:

- The NFSv4 user ID mapping domain name must be the same on the NFSv4 server and target clients.
  It does not necessarily need to be the same as an LDAP or NIS domain name as long as the NFSv4 server and clients are using the same name.
- Target clients must support the NFSv4 numeric ID setting.
- For security reasons, you should use LDAP for name services in NFSv4 deployments.

Steps

1. Verify that NFS is licensed on your cluster:

   system license show -package nfs

   If it is not, contact your sales representative.

2. Create an NFS server:

   vserver nfs create -vserver *vserver_name* -v3 {enabled|disabled} -v4.0 {enabled|disabled} -v4-id-domain *nfsv4_id_domain* -v4-numeric-ids {enabled|disabled} -v4.1 {enabled|disabled} -v4.1-pnfs {enabled|disabled}

   You can choose to enable any combination of NFS versions. If you want to support pNFS, you must enable both `-v4.1` and `-v4.1-pnfs` options.

   If you enable v4 or later, you should also be sure that the following options are set correctly:

   - `-v4-id-domain`
     This optional parameter specifies the domain portion of the string form of user and group names as defined by the NFSv4 protocol. By default, Data ONTAP uses the NIS domain if one is set; if not, the DNS domain is used. You must supply a value that matches the domain name used by target clients.
   - `-v4-numeric-ids`
     This optional parameter specifies whether the support for numeric string identifiers in NFSv4 owner attributes is enabled. The default setting is enabled but you should verify that the target clients support it.

   You can enable additional NFS features later by using the `vserver nfs modify` command.

3. Verify that NFS is running:

   vserver nfs status -vserver *vserver_name*

4. Verify that NFS is configured as desired:

   vserver nfs show -vserver *vserver_name*

Examples

The following command creates an NFS server on the SVM named vs1 with NFSv3 and NFSv4.0 enabled:

```
vs1::> vserver nfs create -vserver vs1 -v3 enabled -v4.0 enabled -
v4-id-domain my_domain.com
```

The following commands verify the status and configuration values of the new NFS server named vs1:

```
vs1::> vserver nfs status -vserver vs1
The NFS server is running on Vserver "vs1".

vs1::> vserver nfs show -vserver vs1

                            Vserver: vs1
                 General NFS Access: true
                             NFS v3: enabled
                           NFS v4.0: enabled
                       UDP Protocol: enabled
                       TCP Protocol: enabled
               Default Windows User: -
                NFSv4.0 ACL Support: disabled
```

```
    NFSv4.0 Read Delegation Support: disabled
   NFSv4.0 Write Delegation Support: disabled
            NFSv4 ID Mapping Domain: my_domain.com
...
```

Related information

ONTAP 9 man page: vserver nfs create

Creating a LIF

A LIF is an IP address associated with a physical or logical port. If there is a component failure, a LIF can fail over to or be migrated to a different physical port, thereby continuing to communicate with the cluster.

Before you begin

- The underlying physical or logical network port must have been configured to the administrative **up** status.
- If you are planning to use a subnet name to allocate the IP address and network mask value for a LIF, the subnet must already exist.
  Subnets contain a pool of IP addresses that belong to the same layer 3 subnet. They are created using the `network subnet create` command.

About this task

- You can create both IPv4 and IPv6 LIFs on the same network port.
- If you are using Kerberos authentication, enable Kerberos on multiple LIFs.
- If you have large number of LIFs in your cluster, you can verify the LIF capacity supported on the cluster by using the `network interface capacity show` command and the LIF capacity supported on each node by using the `network interface capacity details show` command (at the advanced privilege level).
  ONTAP 9 man page: network interface capacity show
  ONTAP 9 man page: network interface capacity details show

Steps

1. Create a LIF:

   ```
   network interface create -vserver vserver_name -lif lif_name -role data
   -data-protocol nfs -home-node node_name -home-port port_name {-address
   IP_address -netmask IP_address | -subnet-name} -firewall-policy data -
   auto-revert {true|false}
   ```

   - The `-data-protocol` parameter must be specified when the LIF is created, and cannot be modified later without destroying and re-creating the data LIF.
   - `-home-node` is the node to which the LIF returns when the `network interface revert` command is run on the LIF.
     You can also specify whether the LIF should automatically revert to the home-node and home-port with the `-auto-revert` option.
   - `-home-port` is the physical or logical port to which the LIF returns when the `network interface revert` command is run on the LIF.

- You can specify an IP address with the `-address` and `-netmask` options, or you enable allocation from a subnet with the `-subnet_name` option.
- When using a subnet to supply the IP address and network mask, if the subnet was defined with a gateway, a default route to that gateway is added automatically to the SVM when a LIF is created using that subnet.
- If you assign IP addresses manually (without using a subnet), you might need to configure a default route to a gateway if there are clients or domain controllers on a different IP subnet.
  ONTAP 9 man page: network route create
- For the `-firewall-policy` option, use the same default **`data`** as the LIF role.
  You can create and add a custom firewall policy later if desired.

2. Verify that the LIF was created successfully by using the `network interface show` command.

3. Verify that the configured IP address is reachable:

To verify an...	Use...
IPv4 address	`network ping`
IPv6 address	`network ping6`

4. If you are using Kerberos, repeat *Steps 1 through 3* to create additional LIFs.

Kerberos must be enabled separately on each of these LIFs.

Examples

The following command creates a LIF and specifies the IP address and network mask values using the `-address` and `-netmask` parameters:

```
cluster-1::> network interface create -vserver vs1 -lif datalif1 -
role data -data-protocol nfs -home-node node-4 -home-port e1c -
address 192.0.2.145 -netmask 255.255.255.0 -firewall-policy data -
auto-revert true
```

The following command creates a LIF and assigns IP address and network mask values from the specified subnet (named client1_sub):

```
cluster-1::> network interface create -vserver vs3 -lif datalif3 -
role data -data-protocol nfs -home-node node-3 -home-port e1c -
subnet-name client1_sub -firewall-policy data -auto-revert true
```

The following command shows all the LIFs in cluster-1. Data LIFs datalif1 and datalif3 are configured with IPv4 addresses, and datalif4 is configured with an IPv6 address:

```
cluster-1::> network interface show

            Logical    Status     Network          Current      Current Is
Vserver     Interface  Admin/Oper Address/Mask     Node         Port    Home
----------- ---------- ---------- ---------------- ------------ ------- ----
cluster-1
            cluster_mgmt up/up    192.0.2.3/24     node-1       e1a     true
node-1
            clus1        up/up    192.0.2.12/24    node-1       e0a     true
            clus2        up/up    192.0.2.13/24    node-1       e0b     true
            mgmt1        up/up    192.0.2.68/24    node-1       e1a     true
node-2
            clus1        up/up    192.0.2.14/24    node-2       e0a     true
            clus2        up/up    192.0.2.15/24    node-2       e0b     true
            mgmt1        up/up    192.0.2.69/24    node-2       e1a     true
vs1
            datalif1     up/down  192.0.2.145/30   node-1       e1c     true
```

```
vs3
            datalif3     up/up    192.0.2.146/30   node-2       e0c     true
            datalif4     up/up    2001::2/64       node-2       e0c     true
5 entries were displayed.
```

Related tasks

Enabling Kerberos on a data LIF on page 43

Related information

ONTAP 9 man page: network interface create

Enabling DNS for host-name resolution

You can use the `vserver services name-service dns` command to enable DNS on an SVM, and configure it to use DNS for host-name resolution. Host names are resolved using external DNS servers.

Before you begin

A site-wide DNS server must be available for host name lookups.

Steps

1. Enable DNS on the SVM:

`vserver services name-service dns create -vserver` ***`vserver_name`*** **`-domains`** ***`lab.company.com`*** **`-name-servers`** ***`10.19.2.30,10.19.2.32`*** **`-state`** ***`enabled`***

Example

The following command enables external DNS server servers on the SVM vs1:

```
cluster-1::> vserver services name-service dns create -vserver vs1 -
domains lab.company.com -name-servers 10.19.2.30,10.19.2.32 -state
enabled
```

2. Display the DNS domain configurations by using the `vserver services name-service dns show` command.

Example

The following command displays the DNS configurations for all SVMs in the cluster:

```
cluster-1::> vserver services name-service dns show
                                                     Name
Vserver        State     Domains                     Servers
-------------- --------- ------------------------ -------------
cluster1       enabled   xyz.company.com             192.56.0.129,
                                                     192.56.0.130
vs1            enabled   xyz.company.com             192.56.0.129,
                                                     192.56.0.130
vs2            enabled   xyz.company.com             192.56.0.129,
                                                     192.56.0.130
vs3            enabled   xyz.company.com             192.56.0.129,
                                                     192.56.0.130
```

The following command displays detailed DNS configuration information for SVM vs1:

```
cluster-1::> vserver services name-service dns show -vserver vs1
               Vserver: vs1
               Domains: xyz.company.com
          Name Servers: 192.56.0.129, 192.56.0.130
    Enable/Disable DNS: enabled
        Timeout (secs): 2
      Maximum Attempts: 1
```

Related information

ONTAP 9 man page: vserver services name-service dns create

Configuring name services

Depending on the configuration of your storage system, Data ONTAP needs to be able to look up host, user, group, or netgroup information to provide proper access to clients. You must configure name services to enable Data ONTAP to access local or external name services to obtain this information.

You should use a name service such as NIS or LDAP to facilitate name lookups during client authentication. It is best to use LDAP whenever possible for greater security, especially when deploying NFSv4 or later. You should also configure local users and groups in case external name servers are not available.

Name service information must be kept synchronized on all sources.

Choices

- Configuring the name service switch table on page 25
- Configuring local UNIX users and groups on page 26
- Working with netgroups on page 30
- Creating an NIS domain configuration on page 32
- Using LDAP on page 33

Configuring the name service switch table

You must configure the name service switch table correctly to enable Data ONTAP to consult local or external name services to retrieve host, user, group, netgroup, or name mapping information.

Before you begin

You must have decided which name services you want to use for host, user, group, netgroup, or name mapping as applicable to your environment.

If you plan to use netgroups, all IPv6 addresses specified in netgroups must be shortened and compressed as specified in RFC 5952.

About this task

Do not include information sources that are not being used. For example, if NIS is not being used in your environment, do not specify the `-sources nis` option.

Steps

1. Add the necessary entries to the name service switch table:

   vserver services name-service ns-switch create -vserver *vserver_name* -database *database_name* -sources *source_names*

2. Verify that the name service switch table contains the expected entries in the desired order:

`vserver services name-service ns-switch show -vserver` ***`vserver_name`***

If you want to make any corrections, you must use the `vserver services name-service ns-switch modify` or `vserver services name-service ns-switch delete` commands.

Example

The following example creates a new entry in the name service switch table for the SVM vs1 to use the local netgroup file and an external NIS server to look up netgroup information in that order:

```
cluster::> vserver services name-service ns-switch create -vserver
vs1 -database netgroup -sources files,nis
```

After you finish

- You must configure the name services you have specified for the SVM to provide data access.
- If you delete any name service for the SVM, you must remove it from the name service switch table as well.
  The client access to the storage system might not work as expected, if you fail to delete the name service from the name service switch table.

Related information

ONTAP 9 man page: vserver services name-service ns-switch create

Configuring local UNIX users and groups

You can use local UNIX users and groups on the SVM for authentication and name mappings. You can create UNIX users and groups manually, or you can load a file containing UNIX users or groups from a uniform resource identifier (URI).

There is a default maximum limit of 32,768 local UNIX user groups and group members combined in the cluster. The cluster administrator can modify this limit.

Choices

- Creating a local UNIX user on page 26
- Loading local UNIX users from a URI on page 27
- Creating a local UNIX group on page 28
- Adding a user to a local UNIX group on page 28
- Loading local UNIX groups from a URI on page 29

Creating a local UNIX user

You can use the `vserver services name-service unix-user create` command to create local UNIX users. A local UNIX user is a UNIX user you create on the SVM as a UNIX name services option to be used in the processing of name mappings.

Step

1. Create a local UNIX user:

   `vserver services name-service unix-user create -vserver` ***`vserver_name`*** **`-user`** ***`user_name`*** **`-id`** ***`integer`*** **`-primary-gid`** ***`integer`*** **`-full-name`** ***`full_name`***

`-user` *`user_name`* specifies the user name. The length of the user name must be 64 characters or fewer.

`-id` *`integer`* specifies the user ID that you assign.

`-primary-gid` *`integer`* specifies the primary group ID. This adds the user to the primary group. After creating the user, you can manually add the user to any desired additional group.

Example

The following command creates a local UNIX user named johnm (full name "John Miller") on the SVM named vs1. The user has the ID 123 and the primary group ID 100.

```
node::> vserver services name-service unix-user create -vserver vs1
-user johnm -id 123
-primary-gid 100 -full-name "John Miller"
```

Related information

ONTAP 9 man page: vserver services name-service unix-user create

Loading local UNIX users from a URI

As an alternative to manually creating individual local UNIX users in SVMs, you can simplify the task by loading a list of local UNIX users into SVMs from a uniform resource identifier (URI) (`vserver services name-service unix-user load-from-uri`).

Steps

1. Create a file containing the list of local UNIX users you want to load.

   The file must contain user information in the UNIX `/etc/passwd` format:
   `user_name``:` *`password`*`:` *`user_ID`*`:` *`group_ID`*`:` *`full_name`*

   The command discards the value of the *`password`* field and the values of the fields after the *`full_name`* field (*`home_directory`* and *`shell`*).

   The maximum supported file size is 2.5 MB.

2. Verify that the list does not contain any duplicate information.

   If the list contains duplicate entries, loading the list fails with an error message.

3. Copy the file to a server.

   The server must be reachable by the storage system over HTTP, HTTPS, FTP, or FTPS.

4. Determine what the URI for the file is.

   The URI is the address you provide to the storage system to indicate where the file is located.

5. Load the file containing the list of local UNIX users into SVMs from the URI:

   `vserver services name-service unix-user load-from-uri -vserver` ***`vserver_name`*** **`-uri {ftp|http|ftps|https}://`*****`uri`*** **`-overwrite {true|false}`**

   `-overwrite` {**`true`**|**`false`**} specifies whether to overwrite entries. The default is **`false`**.

Example

The following command loads a list of local UNIX users from the URI `ftp://ftp.example.com/passwd` into the SVM named vs1. Existing users on the SVM are not overwritten by information from the URI.

```
node::> vserver services name-service unix-user load-from-uri -
vserver vs1
-uri ftp://ftp.example.com/passwd -overwrite false
```

Related information

ONTAP 9 man page: vserver services name-service unix-user load-from-uri

Creating a local UNIX group

You can use the `vserver services name-service unix-group create` command to create UNIX groups that are local to the SVM. Local UNIX groups are used with local UNIX users.

Step

1. Create a local UNIX group:

   ```
   vserver services name-service unix-group create -vserver vserver_name -
   name group_name -id integer
   ```

   `-name group_name` specifies the group name. The length of the group name must be 64 characters or fewer.

   `-id integer` specifies the group ID that you assign.

Example

The following command creates a local group named eng on the SVM named vs1. The group has the ID 101.

```
vs1::> vserver services name-service unix-group create -vserver vs1
-name eng -id 101
```

Related information

ONTAP 9 man page: vserver services name-service unix-group create

Adding a user to a local UNIX group

You can use the `vserver services name-service unix-group adduser` command to add a user to a supplemental UNIX group that is local to the SVM.

Step

1. Add a user to a local UNIX group:

   ```
   vserver services name-service unix-group adduser -vserver vserver_name -
   name group_name -username user_name
   ```

   `-name group_name` specifies the name of the UNIX group to add the user to in addition to the user's primary group.

Example

The following command adds a user named max to a local UNIX group named eng on the SVM named vs1:

```
vs1::> vserver services name-service unix-group adduser -vserver
vs1 -name eng
-username max
```

Related information

ONTAP 9 man page: vserver services name-service unix-group adduser

Loading local UNIX groups from a URI

As an alternative to manually creating individual local UNIX groups, you can load a list of local UNIX groups into SVMs from a uniform resource identifier (URI) by using the `vserver services name-service unix-group load-from-uri` command.

Steps

1. Create a file containing the list of local UNIX groups you want to load.

   The file must contain group information in the UNIX `/etc/group` format:
   `group_name``:` *`password`*`:` *`group_ID`*`:` *`comma_separated_list_of_users`*

   The command discards the value of the *`password`* field.

   The maximum supported file size is 1 MB.

   The maximum length of each line in the group file is 32,768 characters.

2. Verify that the list does not contain any duplicate information.

   The list must not contain duplicate entries, or else loading the list fails. If there are entries already present in the SVM, you must either set the `-overwrite` parameter to **`true`** to overwrite all existing entries with the new file, or ensure that the new file does not contain any entries that duplicate existing entries.

3. Copy the file to a server.

   The server must be reachable by the storage system over HTTP, HTTPS, FTP, or FTPS.

4. Determine what the URI for the file is.

   The URI is the address you provide to the storage system to indicate where the file is located.

5. Load the file containing the list of local UNIX groups into the SVM from the URI:

   `vserver services name-service unix-group load-from-uri -vserver` ***`vserver_name`*** **`-uri {ftp|http|ftps|https}://`*****`uri`*** **`-overwrite {true|false}`**

   `-overwrite` {**`true`**|**`false`**} specifies whether to overwrite entries. The default is **`false`**. If you specify this parameter as **`true`**, Data ONTAP replaces the entire existing local UNIX group database of the specified SVM with the entries from the file you are loading.

Example

The following command loads a list of local UNIX groups from the URI `ftp://ftp.example.com/group` into the SVM named vs1. Existing groups on the SVM are not overwritten by information from the URI.

```
vs1::> vserver services name-service unix-group load-from-uri -
vserver vs1
-uri ftp://ftp.example.com/group -overwrite false
```

Related information

ONTAP 9 man page: vserver services name-service unix-group load-from-uri

Working with netgroups

You can use netgroups for user authentication and to match clients in export policy rules. You can provide access to netgroups from external name servers (LDAP or NIS), or you can load netgroups from a uniform resource identifier (URI) into SVMs using the `vserver services name-service netgroup load` command.

Before you begin

- All hosts in netgroups, regardless of source (NIS, LDAP, or local files), must have both forward (A) and reverse (PTR) DNS records to provide consistent forward and reverse DNS lookups.

  In addition, if an IP address of a client has multiple PTR records, all of those host names must be members of the netgroup and have corresponding A records.

- All IPv6 addresses specified in netgroups must be shortened and compressed as specified in RFC 5952.

  For example, 2011:hu9:0:0:0:0:3:1 must be shortened to 2011:hu9::3:1.

About this task

- You can use the `vserver export-policy netgroup check-membership` command to help determine whether a client IP is a member of a certain netgroup.
- You can use the `vserver services name-service getxxbyyy netgrp` commnd to check whether a client is part of a netgroup.

  The underlying service for doing the lookup is selected based on the configured name service switch order.

Related information

ONTAP 9 man page: vserver export-policy netgroup check-membership

ONTAP 9 man page: vserver services name-service getxxbyyy netgrp

Loading netgroups into SVMs

One of the methods you can use to match clients in export policy rules is by using hosts listed in netgroups. You can load netgroups from a uniform resource identifier (URI) into SVMs as an alternative to using netgroups stored in external name servers (`vserver services name-service netgroup load`).

Before you begin

Netgroup files must meet the following requirements before being loaded into an SVM:

- The file must use the same proper netgroup text file format that is used to populate NIS.

  Data ONTAP checks the netgroup text file format before loading it. If the file contains errors, it will not be loaded and a message is displayed indicating the corrections you have to perform in the file. After correcting the errors, you can reload the netgroup file into the specified SVM.

- Any alphabetic characters in host names in the netgroup file should be lowercase.
- The maximum supported file size is 5 MB.
- The maximum supported level for nesting netgroups is 1000.
- Only primary DNS host names can be used when defining host names in the netgroup file.

To avoid export access issues, host names should not be defined using DNS CNAME or round robin records.

- The user and domain portions of triples in the netgroup file should be kept empty because Data ONTAP does not support them.
  Only the host/IP part is supported.

About this task

Data ONTAP supports netgroup-by-host searches for the local netgroup file. After you load the netgroup file, Data ONTAP automatically creates a netgroup.byhost map to enable netgroup-by-host searches. This can significantly speed up local netgroup searches when processing export policy rules to evaluate client access.

Step

1. Load netgroups into SVMs from a URI:

   `vserver services name-service netgroup load -vserver` ***`vserver_name`*** **`-source {ftp|http|ftps|https}://`*****`uri`***

   Loading the netgroup file and building the netgroup.byhost map can several minutes.

   If you want to update the netgroups, you can edit the file and load the updated netgroup file into the SVM.

Example

The following command loads netgroup definitions into the SVM named vs1 from the HTTP URL `http://intranet/downloads/corp-netgroup`:

```
vs1::> vserver services name-service netgroup load -vserver vs1
-source http://intranet/downloads/corp-netgroup
```

Related information

ONTAP 9 man page: vserver services name-service netgroup load

Verifying the status of netgroup definitions

After loading netgroups into the SVM, you can use the `vserver services name-service netgroup status` command to verify the status of netgroup definitions. This enables you to determine whether netgroup definitions are consistent on all of the nodes that back the SVM.

Steps

1. Set the privilege level to advanced:

   `set -privilege advanced`

2. Verify the status of netgroup definitions:

   `vserver services name-service netgroup status`

   You can display additional information in a more detailed view.

3. Return to the admin privilege level:

   `set -privilege admin`

Example

After the privilege level is set, the following command displays netgroup status for all SVMs:

```
vs1::> set -privilege advanced

Warning: These advanced commands are potentially dangerous; use them only when
         directed to do so by technical support.
Do you wish to continue? (y or n): y

vs1::*> vserver services name-service netgroup status
Virtual
Server    Node            Load Time           Hash Value
--------- --------------- ------------------- --------------------------------
vs1
          node1           9/20/2006 16:04:53  e6cb38ec1396a280c0d2b77e3a84eda2
          node2           9/20/2006 16:06:26  e6cb38ec1396a280c0d2b77e3a84eda2
          node3           9/20/2006 16:08:08  e6cb38ec1396a280c0d2b77e3a84eda2
          node4           9/20/2006 16:11:33  e6cb38ec1396a280c0d2b77e3a84eda2
```

Related information

ONTAP 9 man page: vserver services name-service netgroup status

Creating an NIS domain configuration

If a Network Information Service (NIS) is used in your environment for name services, you must create an NIS domain configuration for the SVM by using the `vserver services name-service nis-domain create` command.

Before you begin

All configured NIS servers must be available and reachable before you configure the NIS domain on the SVM.

If you plan to use NIS for directory searches, the maps in your NIS servers cannot have more than 1,024 characters for each entry. Do not specify the NIS server that does not comply with this limit. Otherwise, client access dependent on NIS entries might fail.

About this task

You can create multiple NIS domains. However, you can only use one that is set to **`active`**.

If your NIS database contains a `netgroup.byhost` map, Data ONTAP can use it for quicker searches. The `netgroup.byhost` and `netgroup` maps in the directory must be kept in sync at all times to avoid client access issues.

Using NIS for host name resolution is not supported.

Steps

1. Create an NIS domain configuration:

   `vserver services name-service nis-domain create -vserver vs1 -domain` ***`domain_name`*** **`-active true -servers`** ***`IP_addresses`***

   You can specify up to 10 NIS servers.

2. Verify that the domain is created:

   `vserver services name-service nis-domain show`

Example

The following command creates and makes an active NIS domain configuration for an NIS domain called nisdomain on the SVM named vs1 with an NIS server at IP address 192.0.2.180:

```
vs1::> vserver services name-service nis-domain create -vserver vs1
-domain nisdomain -active true -servers 192.0.2.180
```

Related information

ONTAP 9 man page: vserver services name-service nis-domain create

Using LDAP

If LDAP is used in your environment for name services, you need to work with your LDAP administrator to determine requirements and appropriate storage system configurations, then enable the SVM as an LDAP client.

- Before configuring LDAP for clustered Data ONTAP, you should verify that your site deployment meets best practices for LDAP server and client configuration. In particular, the following conditions must be met.
    - The domain name of the LDAP server must match the entry on the LDAP client.
    - If the LDAP server requires session security measures, you must configure them in the LDAP client.
      The following session security options are available.
        - LDAP signing (provides data integrity checking) and LDAP signing and sealing (provides data integrity checking and encryption)
        - LDAP over TLS (encryption)
    - To enable signed and sealed LDAP queries, the following services must be configured.
        - LDAP servers must support the GSSAPI (Kerberos) SASL mechanism.
        - LDAP servers must have DNS A/AAAA records as well as PTR records set up on the DNS server.
        - Kerberos servers must have SRV records present on the DNS server.
    - To enable TLS encrypted LDAP queries, the following services must be configured.
        - The LDAP server must be enabled for TLS.
          As of ONTAP 9.0, SSL is no longer supported.
        - A certificate server must already be configured in the domain.
- You must enter an LDAP schema when configuring the LDAP client on the SVM.
  In most cases, one of the default Data ONTAP schemas will be appropriate. However, if the LDAP schema in your environment differs from these, you must create a new LDAP client schema for Data ONTAP before creating the LDAP client. Consult with your LDAP administrator about requirements for your environment.
- Using LDAP for host name resolution is not supported.

Steps

1. Creating a new LDAP client schema on page 34
2. Installing the self-signed root CA certificate on the SVM on page 34
3. Creating an LDAP client configuration on page 35
4. Associating the LDAP client configuration with SVMs on page 37
5. Verifying LDAP sources in the name service switch table on page 38

Creating a new LDAP client schema

Data ONTAP provides three LDAP schemas: one for Active Directory Services for UNIX compatibility, one for Active Directory Identity Management for UNIX compatibility, and one for RFC-2307 LDAP compatibility. If the LDAP schema in your environment differs from these, you must create a new LDAP client schema for Data ONTAP before creating the LDAP client configuration.

About this task

Consult with your LDAP administrator before creating a new schema.

If you need to use a non-default LDAP schema, you must create it before creating the LDAP client configuration.

The default LDAP schemas provided by Data ONTAP cannot be modified. To create a new schema, you create a copy and then modify the copy accordingly.

Steps

1. Display the existing LDAP client schema templates to identify the one you want to copy:

   `vserver services name-service ldap client schema show`

2. Set the privilege level to advanced:

   `set -privilege advanced`

3. Make a copy of an existing LDAP client schema:

   `vserver services name-service ldap client schema copy -vserver` ***`vserver_name`*** **`-schema`** ***`existing_schema_name`*** **`-new-schema-name`** ***`new_schema_name`***

4. Modify the new schema and customize it for your environment:

   `vserver services name-service ldap client schema modify`

5. Return to the admin privilege level:

   `set -privilege admin`

Related information

ONTAP 9 man page: vserver services name-service ldap client schema copy
ONTAP 9 man page: vserver services name-service ldap client schema modify

Installing the self-signed root CA certificate on the SVM

If LDAP authentication with TLS is required when binding to LDAP servers, you must first install the self-signed root CA certificate on the SVM.

About this task

When LDAP over TLS is enabled, the ONTAP LDAP client on the SVM does not support revoked certificates. The LDAP client treats revoked certificates as if they are not revoked.

Steps

1. Install the self-signed root CA certificate:

   a. Begin the certificate installation:

      `security certificate install -vserver` ***`vserver_name`*** **`-type server-ca`**

The console output displays the following message:
`Please enter Certificate: Press <Enter> when done`

b. Open the certificate `.pem` file with a text editor, copy the certificate, including the lines beginning with `-----BEGIN CERTIFICATE-----` and ending with `-----END CERTIFICATE-----`, and then paste the certificate after the command prompt.

c. Verify that the certificate is displayed correctly.

d. Complete the installation by pressing Enter.

2. Verify that the certificate is installed:

```
security certificate show -vserver vserver_name
```

Related information

ONTAP 9 man page: security certificate install

Creating an LDAP client configuration

If you want ONTAP to access external LDAP servers in your environment, you must first set up an LDAP client on the storage system. To do so, you need to gather configuration values for the LDAP server, and then you can use the `vserver services name-service ldap client create` command to create an LDAP client configuration on an SVM.

Steps

1. Consult with your LDAP administrator to determine the appropriate configuration values for the `vserver services name-service ldap client create` command:

a. Specify a domain-based or address-based connection to LDAP servers.

The `-ad-domain` and `-servers` options are mutually exclusive.

- Use the `-ad-domain` option to enable LDAP server discovery in the Active Directory domain.
  You can use the `-preferred-ad-servers` option to specify one or more preferred Active Directory servers by IP address in a comma-delimited list. After the client is created, you can modify this list using the `vserver services name-service ldap client modify` command.

- Use the `-servers` option to specify one or more LDAP servers (AD or UNIX) by IP address in a comma-delimited list.

b. Specify a default or custom LDAP schema.

Most LDAP servers can use the default read-only schemas provided by clustered ONTAP. It is best to use those default schemas unless there is a requirement to do otherwise. If so, you can create your own schema by copying a default schema (they are read-only) and modifying the copy.

Default schemas:

- **`AD-IDMU`**
  Based on Active Directory Identity Management for UNIX, it is appropriate for most Windows 2008, Windows 2012 and later AD servers.

- **`AD-SFU`**
  Based on Active Directory Services for UNIX, it is appropriate for most Windows 2003 and earlier AD servers.

- **`RFC-2307`**

Based on RFC-2307 (*An Approach for Using LDAP as a Network Information Service*), it is appropriate for most UNIX AD servers.

c. Select bind values.

- `-min-bind-level` {**`anonymous`**|**`simple`**|**`sasl`**} specifies the minimum bind authentication level.
  The default is **`anonymous`**.

- `-bind-dn` *`LDAP_DN`* specifies the bind user.
  For Active Directory servers, specify the user in the account (DOMAIN\user) or principal (user@domain.com) form. Otherwise, specify the user in distinguished name (CN=user,DC=domain,DC=com) form.

- `-bind-password` *`password`* specifies the bind password.

d. Select session security options, if required

You can enable either LDAP signing and sealing or LDAP over TLS if required by the LDAP server.

- `--session-security` {**`none`**|**`sign`**|**`seal`**}
  You can enable signing (**`sign`**, data integrity), signing and sealing (**`seal`**, data integrity and encryption), or neither (**`none`**, no signing or sealing). The default value is **`none`**.
  You should also set `-min-bind-level` {**`sasl`**} unless you want the bind authentication to fall back to **`anonymous`** or **`simple`** if the signing and sealing bind fails.

- `-use-start-tls` {**`true`**|**`false`**}
  If set to **`true`** and the LDAP server supports it, the LDAP client uses an encrypted TLS connection to the server. The default is **`false`**. You must install a self-signed root CA certificate of the LDAP server to use this option.

Note: If the SVM has a CIFS server added to a domain and the LDAP server is one of the domain controllers of the home-domain of the CIFS server, then you can modify the `-session-security-for-ad-ldap` option by using the `vserver cifs security modify` command.

e. Select port, query, and base values.

The default values are recommended, but verify with your LDAP administrator that they are appropriate for your environment.

- `-port` *`port`* specifies the LDAP server port.
  The default value is **`389`**.
  If you plan to use Start TLS to secure the LDAP connection, you must use the default port 389. Start TLS begins as a plaintext connection over the LDAP default port 389, and that connection is then upgraded to TLS. If you change the port, Start TLS fails.

- `-query-timeout` *`integer`* specifies the query timeout in seconds.
  The allowed range is from 0 through 10 seconds. The default value is **`3`** seconds.

- `-base-dn` *`LDAP_DN`* specifies the base DN.
  The default value is **`""`** (root).

- `-base-scope` {**`base`**|**`onelevel`**|**`subtree`**} specifies the base search scope.
  The default value is **`subtree`**.

2. Create an LDAP client configuration on the SVM:

```
vserver services name-service ldap client create -vserver vserver_name -
client-config client_config_name {-servers LDAP_server_list | -ad-domain
ad_domain -preferred-ad-servers preferred_ad_server_list -schema schema
```

```
-port 389 -query-timeout 3 -min-bind-level {anonymous|simple|sasl} -
bind-dn LDAP_DN -bind-password password -base-dn LDAP_DN -base-scope
subtree -session-security {none|sign|seal}
```

Note: You must provide the SVM name when creating an LDAP client configuration.

3. Verify that the LDAP client configuration is created successfully:

```
vserver services name-service ldap client show -client-config
client_config_name
```

Examples

The following command creates a new LDAP client configuration named ldap1 for the SVM vs1 to work with an Active Directory server for LDAP:

```
cluster1::> vserver services name-service ldap client create -
vserver vs1 -client-config ldapclient1 –ad-domain
addomain.example.com -schema AD-SFU -port 389 -query-timeout 3 -min-
bind-level simple -base-dn DC=addomain,DC=example,DC=com -base-
scope subtree -preferred-ad-servers 172.17.32.100
```

The following command creates a new LDAP client configuration named ldap1 for the SVM vs1 to work with an Active Directory server for LDAP on which signing and sealing is required:

```
cluster1::> vserver services name-service ldap client create -
vserver vs1 -client-config ldapclient1 –ad-domain
addomain.example.com -schema AD-SFU -port 389 -query-timeout 3 -min-
bind-level sasl -base-dn DC=addomain,DC=example,DC=com -base-scope
subtree -preferred-ad-servers 172.17.32.100 -session-security seal
```

The following command modifies the LDAP client configuration named ldap1 for the SVM vs1 by specifying the base DN:

```
cluster1::> vserver services name-service ldap client modify -
vserver vs1 -client-config ldap1 -base-dn
CN=Users,DC=addomain,DC=example,DC=com
```

Related information

ONTAP 9 man page: vserver services name-service ldap client create

Associating the LDAP client configuration with SVMs

To enable LDAP on an SVM, you must use the `vserver services name-service ldap create` command to associate an LDAP client configuration with the SVM.

Before you begin

- An LDAP domain must already exist within the network and must be accessible to the cluster that the SVM is located on.
- An LDAP client configuration must exist on the SVM.

Step

1. Enable LDAP on the SVM:

vserver services name-service ldap create -vserver *vserver_name* -client-config *client_config_name*

Example

The following command enables LDAP on the “vs1” SVM and configures it to use the “ldap1” LDAP client configuration:

```
cluster1::> vserver services name-service ldap create -vserver vs1 -
client-config ldap1 -client-enabled true
```

Related information

ONTAP 9 man page: vserver services name-service ldap create

Verifying LDAP sources in the name service switch table

You must verify that LDAP sources for name services are listed correctly in the name service switch table for the SVM.

Steps

1. Display the current name service switch table contents:

   vserver services name-service ns-switch show -vserver *svm_name*

   #### Example

   The following command shows the results for the SVM My_SVM:

   ```
   ie3220-a::> vserver services name-service ns-switch show -vserver
   My_SVM
                                  Source
   Vserver         Database       Order
   --------------- ------------   ---------
   My_SVM          hosts          files,
                                  dns
   My_SVM          group          files,ldap
   My_SVM          passwd         files,ldap
   My_SVM          netgroup       files
   My_SVM          namemap        files
   5 entries were displayed.
   ```

   `namemap` specifies the sources to search for name mapping information and in what order. In a UNIX-only environment, this entry is not necessary. Name mapping is only required in a mixed environment using both UNIX and Windows.

2. Update the `ns-switch` entry as appropriate:

If you want to update the ns-switch entry for...	Enter the command...
User information	**vserver services name-service ns-switch modify -vserver *vserver_name* -database passwd -sources ldap,files**
Group information	**vserver services name-service ns-switch modify -vserver *vserver_name* -database group -sources ldap,files**

If you want to update the ns-switch entry for...	Enter the command...
Netgroup information	**`vserver services name-service ns-switch modify -vserver`** ***`vserver_name`*** **`-database netgroup -sources ldap,files`**

Related information

ONTAP 9 man page: vserver services name-service ns-switch modify

Using Kerberos with NFS for strong security

If Kerberos is used in your environment for strong authentication, you need to work with your Kerberos administrator to determine requirements and appropriate storage system configurations, and then enable the SVM as a Kerberos client.

Your environment should meet the following guidelines:

- Your site deployment should follow best practices for Kerberos server and client configuration before you configure Kerberos for clustered Data ONTAP.
- If possible, use NFSv4 or later if Kerberos authentication is required.
  NFSv3 can be used with Kerberos. However, the full security benefits of Kerberos are only realized in clustered Data ONTAP deployments of NFSv4 or later.
- To promote redundant server access, Kerberos should be enabled on several data LIFs on multiple nodes in the cluster using the same SPN.
- When Kerberos is enabled on the SVM, one of the following security methods must be specified in export rules for volumes or qtrees depending on your NFS client configuration.
  - **`krb5`** (Kerberos v5 protocol)
  - **`krb5i`** (Kerberos v5 protocol with integrity checking using checksums)
  - **`krb5p`** (Kerberos v5 protocol with privacy service)

In addition to the Kerberos server and clients, the following external services must be configured for clustered Data ONTAP to support Kerberos:

- Directory service
  You should use a secure directory service in your environment, such as Active Directory or OpenLDAP, that is configured to use LDAP over SSL/TLS. Do not use NIS, whose requests are sent in clear text and are hence not secure.
- NTP
  You must have a working time server running NTP. This is necessary to prevent Kerberos authentication failure due to time skew.
- Domain name resolution (DNS)
  Each UNIX client and each SVM LIF must have a proper service record (SRV) registered with the KDC under forward and reverse lookup zones. All participants must be properly resolvable via DNS.

Steps

1. Verifying permissions for Kerberos configuration on page 40
2. Creating an NFS Kerberos realm configuration on page 41
3. Configuring NFS Kerberos permitted encryption types on page 42

4. Enabling Kerberos on a data LIF on page 43

Verifying permissions for Kerberos configuration

Kerberos requires that certain UNIX permissions be set for the SVM root volume and for local users and groups.

Steps

1. Display the relevant permissions on the SVM root volume:

   ```
   volume show -volume root_vol_name-fields user,group,unix-permissions
   ```

   The root volume of the SVM must have the following configuration:

Name...	Setting...
UID	root or ID 0
GID	root or ID 0
UNIX permissions	755

   If these values are not shown, use the `volume modify` command to update them.

2. Display the local UNIX users:

   ```
   vserver services name-service unix-user show -vserver vserver_name
   ```

   The SVM must have the following UNIX users configured:

User name	User ID	Primary group ID	Comment
nfs	500	0	Required for GSS INIT phase. The first component of the NFS client user SPN is used as the user. The nfs user is not required if a Kerberos-UNIX name mapping exists for the SPN of the NFS client user.
root	0	0	Required for mounting.

   If these values are not shown, you can use the `vserver services name-service unix-user modify` command to update them.

3. Display the local UNIX groups:

   ```
   vserver services name-service unix-group show -vserver vserver_name
   ```

   The SVM must have the following UNIX groups configured:

Group name	Group ID
daemon	1
root	0

   If these values are not shown, you can use the `vserver services name-service unix-group modify` command to update them.

Creating an NFS Kerberos realm configuration

If you want Data ONTAP to access external Kerberos servers in your environment, you must first configure the SVM to use an existing Kerberos realm. To do so, you need to gather configuration values for the Kerberos KDC server, and then use the `vserver nfs kerberos realm create` command to create the Kerberos realm configuration on an SVM.

Before you begin

The cluster administrator should have configured NTP on the storage system, client, and KDC server to avoid authentication issues. Time differences between a client and server (clock skew) are a common cause of authentication failures.

Steps

1. Consult with your Kerberos administrator to determine the appropriate configuration values to supply with the `vserver nfs kerberos realm create` command.
2. Create a Kerberos realm configuration on the SVM:

   ```
   vserver nfs kerberos realm create -vserver vserver_name -realm
   realm_name {AD_KDC_server_values |AD_KDC_server_values} -comment "text"
   ```

3. Verify that the Kerberos realm configuration was created successfully:

   ```
   vserver nfs kerberos realm show
   ```

Examples

The following command creates an NFS Kerberos realm configuration for the SVM vs1 that uses a Microsoft Active Directory server as the KDC server. The Kerberos realm is AUTH.EXAMPLE.COM. The Active Directory server is named ad-1 and its IP address is 10.10.8.14. The permitted clock skew is 300 seconds (the default). The IP address of the KDC server is 10.10.8.14, and its port number is 88 (the default). “Microsoft Kerberos config” is the comment.

```
vs1::> vserver nfs kerberos realm create -vserver vs1 -realm
AUTH.EXAMPLE.COM -adserver-name ad-1
-adserver-ip 10.10.8.14 -clock-skew 300 -kdc-ip 10.10.8.14 -kdc-
port 88 -kdc-vendor Microsoft
-comment "Microsoft Kerberos config"
```

The following command creates an NFS Kerberos realm configuration for the SVM vs1 that uses an MIT KDC. The Kerberos realm is SECURITY.EXAMPLE.COM. The permitted clock skew is 300 seconds. The IP address of the KDC server is 10.10.9.1, and its port number is 88. The KDC vendor is Other to indicate a UNIX vendor. The IP address of the administrative server is 10.10.9.1, and its port number is 749 (the default). The IP address of the password server is 10.10.9.1, and its port number is 464 (the default). “UNIX Kerberos config” is the comment.

```
vs1::> vserver nfs kerberos realm create -vserver vs1 -realm
SECURITY.EXAMPLE.COM. -clock-skew 300
-kdc-ip 10.10.9.1 -kdc-port 88 -kdc-vendor Other -adminserver-ip
10.10.9.1 -adminserver-port 749
-passwordserver-ip 10.10.9.1 -passwordserver-port 464 -comment
"UNIX Kerberos config"
```

Related information

ONTAP 9 man page: vserver nfs kerberos realm create

Configuring NFS Kerberos permitted encryption types

By default, Data ONTAP supports the following encryption types for NFS Kerberos: DES, 3DES, AES-128, and AES-256. You can configure the permitted encryption types for each SVM to suit the security requirements for your particular environment by using the `vserver nfs modify` command with the `-permitted-enc-types` parameter.

About this task

For greatest client compatibility, Data ONTAP supports both weak DES and strong AES encryption by default. This means, for example, that if you want to increase security and your environment supports it, you can use this procedure to disable DES and 3DES and require clients to use only AES encryption.

You should use the strongest encryption available. For clustered Data ONTAP 8.3 and later, that is AES-256. You should confirm with your KDC administrator that this encryption level is supported in your environment.

- Enabling or disabling AES entirely (both AES-128 and AES-256) on SVMs is disruptive because it destroys the original DES principal/keytab file, thereby requiring that the Kerberos configuration be disabled on all LIFs for the SVM.
  Before making this change, you should verify that NFS clients do not rely on AES encryption on the SVM.
- Enabling or disabling DES or 3DES does not require any changes to the Kerberos configuration on LIFs.

Step

1. Enable or disable the permitted encryption type you want:

If you want to enable or disable...	Follow these steps...
DES or 3DES	**a.** Configure the NFS Kerberos permitted encryption types of the SVM: **`vserver nfs modify -vserver`** ***`vserver_name`*** **`-permitted-enc-types`** ***`encryption_types`*** Separate multiple encryption types with a comma. **b.** Verify that the change was successful: **`vserver nfs show -vserver`** ***`vserver_name`*** **`-fields permitted-enc-types`**

If you want to enable or disable...	Follow these steps...

AES-128 or AES-256	**a.** Identify on which SVM and LIF Kerberos is enabled:
	`vserver nfs kerberos interface show`
	b. Disable Kerberos on all LIFs on the SVM whose NFS Kerberos permitted encryption type you want to modify:
	`vserver nfs kerberos interface disable -lif lif_name`
	c. Configure the NFS Kerberos permitted encryption types of the SVM:
	`vserver nfs modify -vserver vserver_name -permitted-enc-types encryption_types`
	Separate multiple encryption types with a comma.
	d. Verify that the change was successful:
	`vserver nfs show -vserver vserver_name -fields permitted-enc-types`
	e. Reenable Kerberos on all LIFs on the SVM:
	`vserver nfs kerberos interface enable -lif lif_name -spn service_principal_name`
	f. Verify that Kerberos is enabled on all LIFs:
	`vserver nfs kerberos interface show`

Related information

ONTAP 9 man page: vserver nfs kerberos interface enable
ONTAP 9 man page: vserver nfs modify

Enabling Kerberos on a data LIF

You can use the `vserver nfs kerberos interface enable` command to enable Kerberos on a data LIF. This enables the SVM to use Kerberos security services for NFS.

About this task

If you are using an Active Directory KDC, the first 15 characters of any SPNs used must be unique across SVMs within a realm or domain.

Steps

1. Create the NFS Kerberos configuration:

   ```
   vserver nfs kerberos interface enable -vserver vserver_name -lif
   logical_interface -spn service_principal_name
   ```

   Data ONTAP requires the secret key for the SPN from the KDC to enable the Kerberos interface.

   For Microsoft KDCs, the KDC is contacted and a user name and password prompt are issued at the CLI to obtain the secret key. If you need to create the SPN in a different OU of the Kerberos realm, you can specify the optional `-ou` parameter.

   For non-Microsoft KDCs, the secret key can be obtained using one of two methods:

If you...	**You must also include the following parameter with the command...**
Have the KDC administrator credentials to retrieve the key directly from the KDC	`-admin-username` *`kdc_admin_username`*
Do not have the KDC administrator credentials but have a keytab file from the KDC containing the key	`-keytab-uri` {ftp|http}://*`uri`*

2. Verify that Kerberos was enabled on the LIF:

   `vserver nfs kerberos-config show`

3. Repeat steps *1* on page 43 and *2* on page 44 to enable Kerberos on multiple LIFs.

Example

The following command creates and verifies an NFS Kerberos configuration for the SVM named vs1 on the logical interface ves03-d1, with the SPN nfs/ves03-d1.lab.example.com@TEST.LAB.EXAMPLE.COM in the OU lab2ou:

```
vs1::> vserver nfs kerberos interface enable -lif ves03-d1 -vserver
vs2
-spn nfs/ves03-d1.lab.example.com@TEST.LAB.EXAMPLE.COM -ou
"ou=lab2ou"

vs1::>vserver nfs kerberos-config show
        Logical
Vserver Interface Address       Kerberos  SPN
------- --------- -------       ---------
-------------------------------
vs0     ves01-a1
                  10.10.10.30   disabled  -
vs2     ves01-d1
                  10.10.10.40   enabled   nfs/ves03-
d1.lab.example.com@TEST.LAB.EXAMPLE.COM
2 entries were displayed.
```

Related information

ONTAP 9 man page: vserver nfs kerberos interface enable

Adding storage capacity to an NFS-enabled SVM

To add storage capacity to an NFS-enabled SVM, you must create a volume or qtree to provide a storage container, and create or modify an export policy for that container. You can then verify NFS client access from the cluster and test access from client systems.

Before you begin

- NFS must be completely set up on the SVM.
- The default export policy of the SVM root volume must contain a rule that permits access to all clients.
- Any updates to your name services configuration must be complete.
- Any additions or modifications to a Kerberos configuration must be complete.

Steps

1. Creating an export policy on page 45
2. Adding a rule to an export policy on page 46
3. Creating a volume or qtree storage container on page 50
4. Securing NFS access using export policies on page 52
5. Verifying NFS client access from the cluster on page 54
6. Testing NFS access from client systems on page 55

Related concepts

Configuring name services on page 25
Using Kerberos with NFS for strong security on page 39

Related tasks

Opening the export policy of the SVM root volume on page 19

Creating an export policy

Before creating export rules, you must create an export policy to hold them. You can use the `vserver export-policy create` command to create an export policy.

Steps

1. Create an export policy:

   `vserver export-policy create -vserver` ***`vserver_name`*** **`-policyname`** ***`policy_name`***

   The policy name can be up to 256 characters long.

2. Verify that the export policy was created:

   `vserver export-policy show -policyname` ***`policy_name`***

Example

The following commands create and verify the creation of an export policy named exp1 on the SVM named vs1:

```
vs1::> vserver export-policy create -vserver vs1 -policyname exp1

vs1::> vserver export-policy show -policyname exp1
Vserver          Policy Name
---------------  -------------------
vs1              exp1
```

Related information

ONTAP 9 man page: vserver export-policy create

Adding a rule to an export policy

Without rules, the export policy cannot provide client access to data. To create a new export rule, you must identify clients and select a client match format, select the access and security types, specify an anonymous user ID mapping, select a rule index number, and select the access protocol. You can then use the `vserver export-policy rule create` command to add the new rule to an export policy.

Before you begin

- The export policy you want to add the export rules to must already exist.
- DNS must be correctly configured on the data SVM and DNS servers must have correct entries for NFS clients.
  This is because Data ONTAP performs DNS lookups using the DNS configuration of the data SVM for certain client match formats, and failures in export policy rule matching can prevent client data access.
- If you are authenticating with Kerberos, you must have determined which of the following security methods is used on your NFS clients:
  - **`krb5`** (Kerberos V5 protocol)
  - **`krb5i`** (Kerberos V5 protocol with integrity checking using checksums)
  - **`krb5p`** (Kerberos V5 protocol with privacy service)

About this task

It is not necessary to create a new rule if an existing rule in an export policy covers your client match and access requirements.

If you are authenticating with Kerberos and if all volumes of the SVM are accessed over Kerberos, you can set the export rule options **`-rorule`**, **`-rwrule`**, and **`-superuser`** for the root volume to **`krb5`**, **`krb5i`**, or **`krb5p`**.

Steps

1. Identify the clients and the client match format for the new rule.

   The `-clientmatch` option specifies the clients to which the rule applies. Single or multiple client match values can be specified; specifications of multiple values must be separated by commas. You can specify the match in any of the following formats:

Client match format	Example
Domain name preceded by the "." character	**`.example.com`** or **`.example.com,.example.net,...`**
Host name	**`host1`** or **`host1,host2, ...`**
IPv4 address	**`10.1.12.24`** or **`10.1.12.24,10.1.12.25, ...`**
IPv4 address with a subnet mask expressed as a number of bits	**`10.1.12.10/4`** or **`10.1.12.10/4,10.1.12.11/4,...`**
IPv4 address with a network mask	**`10.1.16.0/255.255.255.0`** or **`10.1.16.0/255.255.255.0,10.1.17.0/255.255.255.0,...`**
IPv6 address in dotted format	**`::1.2.3.4`** or **`::1.2.3.4,::1.2.3.5,...`**
IPv6 address with a subnet mask expressed as a number of bits	**`ff::00/32`** or **`ff::00/32,ff::01/32,...`**
A single netgroup with the netgroup name preceded by the @ character	**`@netgroup1`** or **`@netgroup1,@netgroup2,...`**

You can also combine types of client definitions; for example, **`.example.com,@netgroup1`**.

When specifying IP addresses, note the following:

- Entering an IP address range, such as 10.1.12.10-10.1.12.70, is not allowed.
  Entries in this format are interpreted as a text string and treated as a host name.
- When specifying individual IP addresses in export rules for granular management of client access, do not specify IP addresses that are dynamically (for example, DHCP) or temporarily (for example, IPv6) assigned.
  Otherwise, the client loses access when its IP address changes.
- Entering an IPv6 address with a network mask, such as ff::12/ff::00, is not allowed.

2. Select the access and security types for client matches.

   You can specify one or more of the following access modes to clients that authenticate with the specified security types:

   - `-rorule` (read-only access)
   - `-rwrule` (read-write access)
   - `-superuser` (root access)

   Note: A client can only get read-write access for a specific security type if the export rule allows read-only access for that security type as well. If the read-only parameter is more restrictive for a security type than the read-write parameter, the client might not get read-write access. The same is true for superuser access.

   You can specify a comma-separated list of multiple security types for a rule. If you specify the security type as **`any`** or **`never`**, do not specify any other security types. Choose from the following valid security types:

When security type is set to...	A matching client can access the exported data...
`any`	Always, regardless of incoming security type.

When security type is set to...	A matching client can access the exported data...
none	If listed alone, clients with any security type are granted access as anonymous. If listed with other security types, clients with a specified security type are granted access and clients with any other security type are granted access as anonymous.
never	Never, regardless of incoming security type.
krb5	If it is authenticated by Kerberos 5. Authentication only: The header of each request and response is signed.
krb5i	If it is authenticated by Kerberos 5i. Authentication and integrity: The header and body of each request and response is signed.
krb5p	If it is authenticated by Kerberos 5p. Authentication, integrity, and privacy: The header and body of each request and response is signed, and the NFS data payload is encrypted.
ntlm	If it is authenticated by CIFS NTLM.
sys	If it is authenticated by NFS AUTH_SYS.

The recommended security type is **sys**, or if Kerberos is used, **krb5**, **krb5i**, or **krb5p**.

If you are using Kerberos with NFSv3, the export policy rule must allow **-rorule** and **-rwrule** access to **sys** in addition to **krb5**. This is because of the need to allow Network Lock Manager (NLM) access to the export.

3. Specify an anonymous user ID mapping.

   The -anon option specifies a UNIX user ID or user name that is mapped to client requests that arrive with a user ID of 0 (zero), which is typically associated with the user name root. The default value is **65534**. NFS clients typically associate user ID 65534 with the user name nobody (also known as *root squashing*). In clustered Data ONTAP, this user ID is associated with the user pcuser. To disable access by any client with a user ID of 0, specify a value of **65535**.

4. Select the rule index order.

   The -ruleindex option specifies the index number for the rule. Rules are evaluated according to their order in the list of index numbers; rules with lower index numbers are evaluated first. For example, the rule with index number 1 is evaluated before the rule with index number 2.

If you are adding...	Then...
The first rule to an export policy	Enter **1**.
Additional rules to an export policy	**a.** Display existing rules in the policy: **vserver export-policy rule show -instance -policyname *your_policy*** **b.** Select an index number for the new rule depending on the order it should be evaluated.

5. Select the applicable NFS access value: {**`nfs`**|**`nfs3`**|**`nfs4`**}.

`nfs` matches any version, **`nfs3`** and **`nfs4`** match only those specific versions.

6. Create the export rule and add it to an existing export policy:

```
vserver export-policy rule create -vserver vserver_name -policyname
policy_name -ruleindex integer -protocol {nfs|nfs3|nfs4} -clientmatch
{ text | "text,text,..." } -rorule security_type -rwrule security_type -
superuser security_type -anon user_ID
```

7. Display the rules for the export policy to verify that the new rule is present:

```
vserver export-policy rule show -policyname policy_name
```

The command displays a summary for that export policy, including a list of rules applied to that policy. Data ONTAP assigns each rule a rule index number. After you know the rule index number, you can use it to display detailed information about the specified export rule.

8. Verify that the rules applied to the export policy are configured correctly:

```
vserver export-policy rule show -policyname policy_name -vserver
vserver_name -ruleindex integer
```

Examples

The following commands create and verify the creation of an export rule on the SVM named vs1 in an export policy named rs1. The rule has the index number 1. The rule matches any client in the domain eng.company.com and the netgroup @netgroup1. The rule enables all NFS access. It enables read-only and read-write access to users that authenticated with AUTH_SYS. Clients with the UNIX user ID 0 (zero) are anonymized unless authenticated with Kerberos.

```
vs1::> vserver export-policy rule create -vserver vs1 -policyname
exp1 -ruleindex 1 -protocol nfs
-clientmatch "eng.company.com,@netgoup1 -rorule sys -rwrule sys -
anon 65534 -superuser krb5

vs1::> vserver export-policy rule show -policyname nfs_policy
Virtual      Policy         Rule    Access    Client           RO
Server       Name           Index   Protocol  Match            Rule
------------ -------------- ------  --------  ----------------
------
vs1          exp1           1       nfs       eng.company.com, sys
                                              @netgroup1

vs1::> vserver export-policy rule show -policyname exp1 -vserver
vs1 -ruleindex 1

                                    Vserver: vs1
                                Policy Name: exp1
                                 Rule Index: 1
                            Access Protocol: nfs
Client Match Hostname, IP Address, Netgroup, or Domain:
eng.company.com,@netgroup1
                             RO Access Rule: sys
                             RW Access Rule: sys
User ID To Which Anonymous Users Are Mapped: 65534
                   Superuser Security Types: krb5
               Honor SetUID Bits in SETATTR: true
                  Allow Creation of Devices: true
```

The following commands create and verify the creation of an export rule on the SVM named vs2 in an export policy named expol2. The rule has the index number 21. The rule matches clients to members of the netgroup dev_netgroup_main. The rule enables all NFS access. It enables read-only access for users that authenticated with AUTH_SYS and requires Kerberos authentication for read-write and root access. Clients with the UNIX user ID 0 (zero) are denied root access unless authenticated with Kerberos.

```
vs2::> vserver export-policy rule create -vserver vs2 -policyname
expol2 -ruleindex 21 -protocol nfs
-clientmatch @dev_netgroup_main -rorule sys -rwrule krb5 -anon
65535 -superuser krb5

vs2::> vserver export-policy rule show -policyname nfs_policy
Virtual  Policy       Rule    Access    Client              RO
Server   Name         Index   Protocol  Match               Rule
-------- ------------ ------  --------  ------------------  ------
vs2      expol2       21       nfs      @dev_netgroup_main  sys

vs2::> vserver export-policy rule show -policyname expol2 -vserver
vs1 -ruleindex 21

                                    Vserver: vs2
                                Policy Name: expol2
                                 Rule Index: 21
                            Access Protocol: nfs
Client Match Hostname, IP Address, Netgroup, or Domain:
                                             @dev_netgroup_main
                             RO Access Rule: sys
                             RW Access Rule: krb5
User ID To Which Anonymous Users Are Mapped: 65535
                   Superuser Security Types: krb5
               Honor SetUID Bits in SETATTR: true
                  Allow Creation of Devices: true
```

Related information

ONTAP 9 man page: vserver export-policy rule create

Creating a volume or qtree storage container

You can provision storage on a volume or a qtree. If you are creating a qtree, the volume that contains it must already exist.

Choices

- Creating a volume on page 50
- Creating a qtree on page 51

Creating a volume

You can create a volume and specify its junction point and other properties by using the `volume create` command.

Before you begin

The SVM security style must be UNIX, and NFS should be set up and running.

Steps

1. Create the volume with a junction point:

   ```
   volume create -vserver vserver_name -volume volume_name -aggregate
   aggregate_name -size {integer[KB|MB|GB|TB|PB]} -security-style unix -
   user user_name_or_number -group group_name_or_number -junction-path
   junction_path [-policy export_policy_name]
   ```

The choices for `-junction-path` are the following:

- Directly under root; for example, */new_vol*
- Under a new directory (in a new hierarchy); for example, */new_dir/new_vol*
- Under an existing directory (in an existing hierarchy); for example, */existing_dir/new_vol*

If you plan to use an existing export policy, you can specify it when you create the volume. You can also add an export policy later with the `volume modify` command.

2. Verify that the volume was created with the desired junction point:

`volume show -vserver` ***`vserver_name`*** **`-volume`** ***`volume_name`*** **`-junction`**

Example

The following command creates a new volume named home4 on the SVM vs1 and the aggregate aggr1. The volume is made available at `/eng/home` in the namespace for the vs1 SVM. The volume is 750 GB in size, and its volume guarantee is of type volume (by default).

```
cluster1::> volume create -vserver vs1 -volume home4 -aggregate aggr1 -size
750g -junction-path /eng/home
[Job 1642] Job succeeded: Successful

cluster1::> volume show -vserver vs1 -volume home4 -junction
                  Junction                 Junction
Vserver   Volume  Active   Junction Path   Path Source
--------- ------- -------- --------------- -----------
vs1       home4   true     /eng/home       RW_volume
```

Related information

ONTAP 9 man page: volume create

Creating a qtree

You can create a qtree to contain your data and specify its properties by using the `volume qtree create` command.

Before you begin

- The SVM and the volume that will contain the new qtree must already exist.
- The SVM security style must be UNIX, and NFS should be set up and running.

Steps

1. Create the qtree:

`volume qtree create -vserver` ***`vserver_name`*** **`{ -volume`** ***`volume_name`*** **`-qtree`** ***`qtree_name`*** **`| -qtree-path`** ***`qtree path`*** **`} -security-style unix [-policy`** ***`export_policy_name`*** **`]`**

You can specify the volume and qtree as separate arguments or specify the qtree path argument in the format **`/vol/`*****`volume_name/_qtree_name`***.

By default, qtrees inherit the export policies of their parent volume, but they can be configured to use their own. If you plan to use an existing export policy, you can specify it when you create the qtree. You can also add an export policy later with the `volume qtree modify` command.

2. Verify that the qtree was created with the desired junction path:

volume qtree show -vserver ***vserver_name*** **{ -volume** ***volume_name*** **-qtree** ***qtree_name*** **| -qtree-path** ***qtree path*** **}**

Example

The following example creates a qtree named qt01 located on SVM vs1 that has a junction path `/vol/data1`:

```
cluster1::> volume qtree create -vserver vs1 -qtree-path /vol/data1/qt01 -
security-style unix
[Job 1642] Job succeeded: Successful

cluster1::> volume qtree show -vserver vs1 -qtree-path /vol/data1/qt01

                      Vserver Name: vs1
                       Volume Name: data1
                        Qtree Name: qt01
  Actual (Non-Junction) Qtree Path: /vol/data1/qt01
                    Security Style: unix
                       Oplock Mode: enable
                  Unix Permissions: ---rwxr-xr-x
                          Qtree Id: 2
                      Qtree Status: normal
                     Export Policy: default
        Is Export Policy Inherited: true
```

Related information

ONTAP 9 man page: volume qtree create

Securing NFS access using export policies

You can use export policies to restrict NFS access to volumes or qtrees to clients that match specific parameters. When provisioning new storage, you can use an existing policy and rules, add rules to an existing policy, or create a new policy and rules.

Choices

- Managing the processing order of export rules on page 52
- Assigning an export policy to a volume on page 53
- Assigning an export policy to a qtree on page 54

Managing the processing order of export rules

You can use the `vserver export-policy rule setindex` command to manually set an existing export rule's index number. This enables you to specify the precedence by which Data ONTAP applies export rules to client requests.

About this task

If the new index number is already in use, the command inserts the rule at the specified spot and reorders the list accordingly.

Step

1. Modify the index number of a specified export rule:

   vserver export-policy rule setindex -vserver ***virtual_server_name*** **-policyname** ***policy_name*** **-ruleindex** ***integer*** **-newruleindex** ***integer***

Example

The following command changes the index number of an export rule at index number 3 to index number 2 in an export policy named rs1 on the SVM named vs1:

```
vs1::> vserver export-policy rule setindex -vserver vs1
-policyname rs1 -ruleindex 3 -newruleindex 2
```

Related information

ONTAP 9 man page: vserver export-policy rule setindex

Assigning an export policy to a volume

Each volume contained in the SVM must be associated with an export policy that contains export rules for clients to access data in the volume.

About this task

You can associate an export policy to a volume when you create the volume or at any time after you create the volume. You can associate one export policy to the volume, although one policy can be associated to many volumes.

Steps

1. If an export policy was not specified when the volume was created, assign an export policy to the volume:

   ```
   volume modify -vserver vserver_name -volume volume_name -policy
   export_policy_name
   ```

2. Verify that the policy was assigned to the volume:

   ```
   volume show -volume volume_name -fields policy
   ```

Example

The following commands assign the export policy nfs_policy to the volume vol1 on the SVM vs1 and verify the assignment:

```
cluster::> volume modify -v1server vs1 -volume vol1 -policy
nfs_policy

cluster::>volume show -volume vol -fields policy
vserver volume       policy
------- ------------ ----------------
vs1     vol1         nfs_policy
```

Related information

ONTAP 9 man page: volume modify

Assigning an export policy to a qtree

Instead of exporting an entire volume, you can also export a specific qtree on a volume to make it directly accessible to clients. You can export a qtree by assigning an export policy to it. You can assign the export policy either when you create a new qtree or by modifying an existing qtree.

Before you begin

The export policy must exist.

About this task

By default, qtrees inherit the parent export policy of the containing volume if not otherwise specified at the time of creation.

You can associate an export policy to a qtree when you create the qtree or at any time after you create the qtree. You can associate one export policy to the qtree, although one policy can be associated with many qtrees.

Steps

1. If an export policy was not specified when the qtree was created, assign an export policy to the qtree:

   `volume qtree modify -vserver` ***`vserver_name`*** **`-qtree-path /vol/`*****`volume_name/ qtree_name`*** **`-export-policy`** ***`export_policy_name`***

2. Verify that the policy was assigned to the qtree:

   `volume qtree show -qtree` ***`qtree_name`*** **`-fields export-policy`**

Example

The following commands assign the export policy nfs_policy to the qtree qt1 on the SVM vs1 and verify the assignment:

```
cluster::> volume modify -v1server vs1 -qtree-path /vol/vol1/qt1 -
policy nfs_policy

cluster::>volume qtree show -volume vol1 -fields export-policy
vserver volume qtree export-policy
------- ------ ----- -------------
vs1     data1  qt01  nfs_policy
```

Related information

ONTAP 9 man page: volume qtree modify

Verifying NFS client access from the cluster

You can give select clients access to the share by setting UNIX file permissions on a UNIX administration host. You can check client access by using the `vserver export-policy check-access` command, adjusting the export rules as necessary.

Steps

1. On the cluster, check client access to exports by using the `vserver export-policy check-access` command.

The following command checks read/write access for an NFSv3 client with the IP address 1.2.3.4 to the volume home2. The command output shows that the volume uses the export policy `exp-home-dir` and that access is denied.

```
cluster1::> vserver export-policy check-access -vserver vs1 -client-ip 1.2.3.4 -volume
home2 -authentication-method sys -protocol nfs3 -access-type read-write
                                      Policy    Policy     Rule
Path                    Policy        Owner     Owner Type Index  Access
----------------------- ------------- --------- ---------- ------ ----------
/                       default       vs1_root  volume          1 read
/eng                    default       vs1_root  volume          1 read
/eng/home2              exp-home-dir  home2     volume          1 denied

3 entries were displayed.
```

2. Examine the output to determine whether the export policy works as intended and the client access behaves as expected.

   Specifically, you should verify which export policy is used by the volume or qtree and the type of access the client has as a result.

3. If necessary, reconfigure the export policy rules.

Related information

ONTAP 9 man page: vserver export-policy check-access

Testing NFS access from client systems

After you verify NFS access to the new storage object, you should test the configuration by logging in to an NFS administration host and reading data from and writing data to the SVM. You should then repeat the process as a non-root user on a client system.

Before you begin

- The client system must have an IP address that is allowed by the export rule you specified earlier.
- You must have the login information for the root user.

Steps

1. On the cluster, verify the IP address of the LIF that is hosting the new volume:

   network interface show –vserver *svm_name*

2. Log in as the root user to the administration host client system.

3. Change the directory to the mount folder:

   cd /mnt/

4. Create and mount a new folder using the IP address of the SVM:

   a. Create a new folder:

      mkdir /mnt/*folder*

   b. Mount the new volume at this new directory:

      mount -t nfs -o hard *IPAddress*:/*volume_name* /mnt/*folder*

   c. Change the directory to the new folder:

      cd *folder*

Example

The following commands create a folder named test1, mount the vol1 volume at the 192.0.2.130 IP address on the test1 mount folder, and change to the new test1 directory:

```
host# mkdir /mnt/test1
host# mount -t nfs -o hard 192.0.2.130:/vol1 /mnt/test1
host# cd /mnt/test1
```

5. Create a new file, verify that it exists, and write text to it:

   a. Create a test file:

      touch *filename*

   b. Verify that the file exists.:

      ls -l *filename*

   c. Enter:

      cat >*filename*

      Type some text, and then press Ctrl+D to write text to the test file.

   d. Display the content of the test file.

      cat *filename*

   e. Remove the test file:

      rm *filename*

   f. Return to the parent directory:

      cd ..

   Example

   ```
   host# touch myfile1
   host# ls -l myfile1
   -rw-r--r-- 1 root root 0 Sep 18 15:58 myfile1
   host# cat >myfile1
   This text inside the first file
   host# cat myfile1
   This text inside the first file
   host# rm -r myfile1
   host# cd ..
   ```

6. As root, set any desired UNIX ownership and permissions on the mounted volume.

7. On a UNIX client system identified in your export rules, log in as one of the authorized users who now has access to the new volume, and repeat the procedures in steps *3* on page 55 to *5* on page 56 to verify that you can mount the volume and create a file.

Related information

ONTAP 9 man page: network interface show

Where to find additional information

After you have successfully tested NFS client access, you can perform additional NFS configuration or add SAN access. When protocol access is complete, you should protect the root volume of SVM. There are express guides, comprehensive guides, and technical reports to help you achieve these goals.

NFS configuration

You can further configure NFS access using the following comprehensive guides and technical reports:

- *NFS management*
  Describes how to configure and manage file access using NFS.
- *NetApp Technical Report 4067: Clustered Data ONTAP Best Practice and NFS Implementation Guide*
  Serves as an NFSv3 and NFSv4 operational guide, and provides an overview of the clustered Data ONTAP operating system with a focus on NFSv4.
- *NetApp Technical Report 4073: Secure Unified Authentication with NetApp Storage Systems: Kerberos, NFSv4, and LDAP for User Authentication over NFS (with a Focus on Clustered Data ONTAP)*
  Explains how to configure clustered Data ONTAP for use with UNIX-based Kerberos version 5 (krb5) servers for NFS storage authentication and Windows Server Active Directory (AD) as the KDC and Lightweight Directory Access Protocol (LDAP) identity provider.
- *NetApp Technical Report 3580: NFSv4 Enhancements and Best Practices Guide: Data ONTAP Implementation*
  Describes the best practices that should be followed while implementing NFSv4 components on AIX, Linux, or Solaris clients attached to systems running clustered Data ONTAP.

Networking configuration

You can further configure networking features and name services using the following comprehensive guides and technical reports:

- *NFS management*
  Describes how to configure and manage clustered Data ONTAP networking.
- *NetApp Technical Report 4182: Ethernet Storage Design Considerations and Best Practices for Clustered Data ONTAP Configurations*
  Describes the implementation of clustered Data ONTAP network configurations, and provides common network deployment scenarios and best practice recommendations.
- *NetApp Technical Report 4379: Name Services Best Practice Guide Clustered Data ONTAP*
  Explains how to configure LDAP, NIS, DNS, and local file configuration for authentication purposes.

SAN protocol configuration

If you want to provide or modify SAN access to the new SVM, you can use any of the FC or iSCSI configuration express guides, which are available for multiple host operating systems.

NetApp Documentation: Clustered Data ONTAP Express Guides

Root volume protection

After configuring protocols on the SVM, you should ensure that its root volume is protected by using the following express guide:

- *SVM root volume protection express configuration*
  Describes how to quickly create load-sharing mirrors on every node of an ONTAP 9.0 cluster to protect the SVM root volume, which is a NetApp best practice for NAS-enabled SVMs. Also describes how to quickly recover from volume failures or losses by promoting the SVM root volume from a load-sharing mirror.

How ONTAP exports differ from 7-Mode exports

If you are unfamiliar with how ONTAP implements NFS exports, you can compare 7-Mode and ONTAP export configuration tools, as well as sample 7-Mode `/etc/exports` files with clustered policies and rules.

In ONTAP there is no `/etc/exports` file and no `exportfs` command. Instead, you must define an export policy. Export policies enable you to control client access in much the same way as you did in 7-Mode, but give you additional functionality such as the ability to reuse the same export policy for multiple volumes.

Related information

NFS management

NetApp Technical Report 4067: Clustered Data ONTAP NFS Best Practice and Implementation Guide

Comparison of exports in 7-Mode and ONTAP

Exports in ONTAP are defined and used differently than they are in 7-Mode environments.

Areas of difference	7-Mode	ONTAP
How exports are defined	Exports are defined in the `/etc/exports` file.	Exports are defined by creating an export policy within an SVM. An SVM can include more than one export policy.
Scope of export	• Exports apply to a specified file path or qtree. • You must create a separate entry in `/etc/exports` for each file path or qtree. • Exports are persistent only if they are defined in the `/etc/exports` file.	• Export policies apply to an entire volume, including all of the file paths and qtrees contained in the volume. • Export policies can be applied to more than one volume if you want. • All export policies are persistent across system restarts.

Areas of difference	7-Mode	ONTAP
Fencing (specifying different access for specific clients to the same resources)	To provide specific clients different access to a single exported resource, you have to list each client and its permitted access in the `/etc/exports` file.	Export policies are composed of a number of individual export rules. Each export rule defines specific access permissions for a resource and lists the clients that have those permissions. To specify different access for specific clients, you have to create an export rule for each specific set of access permissions, list the clients that have those permissions, and then add the rules to the export policy.
Name aliasing	When you define an export, you can choose to make the name of the export different from the name of the file path. You should use the `-actual` parameter when defining such an export in the `/etc/exports` file.	You can choose to make the name of the exported volume different from the actual volume name. To do this, you must mount the volume with a custom junction path name within the SVM namespace. **Note:** By default, volumes are mounted with their volume name. To customize a volume's junction path name you need to unmount it, rename it, and then remount it.

Examples of ONTAP export policies

You can review example export policies to better understand how export policies work in ONTAP.

Sample ONTAP implementation of a 7-Mode export

The following example shows a 7-Mode export as it appears in the `/etc/export` file:

```
/vol/vol1 -sec=sys,ro=@readonly_netgroup,rw=@readwrite_netgroup1:
@readwrite_netgroup2:@rootaccess_netgroup,root=@rootaccess_netgroup
```

To reproduce this export as a clustered export policy, you have to create an export policy with three export rules, and then assign the export policy to the volume vol1.

Rule	Element	Value
Rule 1	`-clientmatch` (client specification)	**`@readonly_netgroup`**
	`-ruleindex` (position of export rule in the list of rules)	**`1`**
	`-protocol`	**`nfs`**
	`-rorule` (allow read-only access)	**`sys`** (client authenticated with AUTH_SYS)
	`-rwrule` (allow read-write access)	**`never`**
	`-superuser` (allow superuser access)	**`none`** (root *squashed* to anon)
Rule 2	`-clientmatch`	**`@rootaccess_netgroup`**
	`-ruleindex`	**`2`**
	`-protocol`	**`nfs`**
	`-rorule`	**`sys`**
	`-rwrule`	**`sys`**
	`-superuser`	**`sys`**
Rule 3	`-clientmatch`	**`@readwrite_netgroup1,@readwrite_netgroup2`**
	`-ruleindex`	**`3`**
	`-protocol`	**`nfs`**
	`-rorule`	**`sys`**
	`-rwrule`	**`sys`**
	`-superuser`	**`none`**

1. Create an export policy called exp_vol1:

   ```
   vserver export-policy create -vserver NewSVM -policyname exp_vol1
   ```

2. Create three rules with the following parameters to the base command:

   - Base command:

     ```
     vserver export-policy rule create -vserver NewSVM -policyname exp_vol1
     ```

   - Rule parameters:

     ```
     -clientmatch @readonly_netgroup -ruleindex 1 -protocol nfs -rorule sys
     -rwrule never -superuser none
     ```

     ```
     -clientmatch @rootaccess_netgroup -ruleindex 2 -protocol nfs -rorule
     sys -rwrule sys -superuser sys
     ```

     ```
     -clientmatch @readwrite_netgroup1,@readwrite_netgroup2 -ruleindex 3 -
     protocol nfs -rorule sys -rwrule sys -superuser none
     ```

3. Assign the policy to the volume vol1:

   ```
   volume modify -vserver NewSVM -volume vol1 -policy exp_vol1
   ```

Sample consolidation of 7-Mode exports

The following example shows a 7-Mode `/etc/export` file that includes one line for each of 10 qtrees:

```
/vol/vol1/q_1472 -sec=sys,rw=host1519s,root=host1519s
/vol/vol1/q_1471 -sec=sys,rw=host1519s,root=host1519s
/vol/vol1/q_1473 -sec=sys,rw=host1519s,root=host1519s
/vol/vol1/q_1570 -sec=sys,rw=host1519s,root=host1519s
/vol/vol1/q_1571 -sec=sys,rw=host1519s,root=host1519s
/vol/vol1/q_2237 -sec=sys,rw=host2057s,root=host2057s
/vol/vol1/q_2238 -sec=sys,rw=host2057s,root=host2057s
/vol/vol1/q_2239 -sec=sys,rw=host2057s,root=host2057s
/vol/vol1/q_2240 -sec=sys,rw=host2057s,root=host2057s
/vol/vol1/q_2241 -sec=sys,rw=host2057s,root=host2057s
```

In ONTAP, one of two policies is needed for each qtree: one with a rule including `-clientmatch host1519s`, or one with a rule including `-clientmatch host2057s`.

1. Create two export policies called exp_vol1q1 and exp_vol1q2:
   - **`vserver export-policy create -vserver NewSVM -policyname exp_vol1q1`**
   - **`vserver export-policy create -vserver NewSVM -policyname exp_vol1q2`**
2. Create a rule for each policy:
   - **`vserver export-policy rule create -vserver NewSVM -policyname exp_vol1q1 -clientmatch host1519s -rwrule sys -superuser sys`**
   - **`vserver export-policy rule create -vserver NewSVM -policyname exp_vol1q2 -clientmatch host1519s -rwrule sys -superuser sys`**
3. Apply the policies to the qtrees:
   - **`volume qtree modify -vserver NewSVM -qtree-path /vol/vol1/q_1472 -export-policy exp_vol1q1`**
   - [next 4 qtrees...]
   - **`volume qtree modify -vserver NewSVM -qtree-path /vol/vol1/q_2237 -export-policy exp_vol1q2`**
   - [next 4 qtrees...]

If you need to add additional qtrees for those hosts later, you would use the same export policies.

Copyright information

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.277-7103 (October 1988) and FAR 52-227-19 (June 1987).

Trademark information

Active IQ, AltaVault, Arch Design, ASUP, AutoSupport, Campaign Express, Clustered Data ONTAP, Customer Fitness, Data ONTAP, DataMotion, Fitness, Flash Accel, Flash Cache, Flash Pool, FlexArray, FlexCache, FlexClone, FlexGroup, FlexPod, FlexScale, FlexShare, FlexVol, FPolicy, Fueled by SolidFire, GetSuccessful, Helix Design, LockVault, Manage ONTAP, MetroCluster, MultiStore, NetApp, NetApp Insight, OnCommand, ONTAP, ONTAPI, RAID DP, RAID-TEC, SANscreen, SANshare, SANtricity, SecureShare, Simplicity, Simulate ONTAP, Snap Creator, SnapCenter, SnapCopy, SnapDrive, SnapIntegrator, SnapLock, SnapManager, SnapMirror, SnapMover, SnapProtect, SnapRestore, Snapshot, SnapValidator, SnapVault, SolidFire, SolidFire Helix, StorageGRID, SyncMirror, Tech OnTap, Unbound Cloud, and WAFL and other names are trademarks or registered trademarks of NetApp, Inc., in the United States, and/or other countries. All other brands or products are trademarks or registered trademarks of their respective holders and should be treated as such. A current list of NetApp trademarks is available on the web.

http://www.netapp.com/us/legal/netapptmlist.aspx

How to send comments about documentation and receive update notifications

You can help us to improve the quality of our documentation by sending us your feedback. You can receive automatic notification when production-level (GA/FCS) documentation is initially released or important changes are made to existing production-level documents.

If you have suggestions for improving this document, send us your comments by email.

doccomments@netapp.com

To help us direct your comments to the correct division, include in the subject line the product name, version, and operating system.

If you want to be notified automatically when production-level documentation is released or important changes are made to existing production-level documents, follow Twitter account @NetAppDoc.

You can also contact us in the following ways:

- NetApp, Inc., 495 East Java Drive, Sunnyvale, CA 94089 U.S.
- Telephone: +1 (408) 822-6000
- Fax: +1 (408) 822-4501
- Support telephone: +1 (888) 463-8277

Index