Configure NFS access to an SVM

ONTAP 9

NetApp
February 24, 2022

Table of Contents

Configure NFS access to an SVM

Create 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 vs1.example.com -rootvolume root_vs1
-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 vs1.example.com
                                    Vserver: vs1.example.com
                                Vserver Type: data
                             Vserver Subtype: default
                                Vserver UUID: b8375669-19b0-11e5-b9d1-
00a0983d9736
                                Root Volume: root_vs1
                                  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: -
                            QoS Policy Group: -
                                 Config Lock: false
                                IPspace Name: ipspaceA
```

Verify that the NFS protocol is enabled on the SVM

Before you can configure and use NFS on SVMs, you must verify that the protocol is enabled.

About this task

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.

> You cannot add or remove a protocol from a LIF once it is created.

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. If necessary, enable or disable a protocol:
   - To enable the NFS protocol:
     `vserver add-protocols -vserver vserver_name -protocols nfs`
   - To disable a protocol:
     `vserver remove-protocols -vserver vserver_name -protocols protocol_name [,protocol_name,…]`
3. Confirm that the enabled and disabled protocols were updated correctly:

   `vserver show -vserver vserver_name -protocols`

Example

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

```
vs1::> vserver show -vserver vs1.example.com -protocols
Vserver           Allowed Protocols         Disallowed Protocols
-----------       ----------------------    -----------------------
vs1.example.com   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.example.com -protocols nfs
```

Open 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 open 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 access is open to all NFS clients 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`

   The command output should be similar to the following:

```
cluster::> vserver export-policy rule show -vserver vs1.example.com
-policyname default -instance

                                    Vserver: vs1.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 -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.

Create 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.

What you'll need

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, 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
...
```

Create 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 network.

What you'll need

- 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.

- The mechanism for specifying the type of traffic handled by a LIF has changed. For ONTAP 9.5 and earlier, LIFs used roles to specify the type of traffic it would handle. Beginning with ONTAP 9.6, LIFs use service policies to specify the type of traffic it would handle.

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 a 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).
- Beginning with ONTAP 9.7, if other LIFs already exist for the SVM in the same subnet, you do not need to

specify the home port of the LIF. ONTAP automatically chooses a random port on the specified home node in the same broadcast domain as the other LIFs already configured in the same subnet.

Beginning with ONTAP 9.4, FC-NVMe is supported. If you are creating an FC-NVMe LIF you should be aware of the following:

- The NVMe protocol must be supported by the FC adapter on which the LIF is created.
- FC-NVMe can be the only data protocol on data LIFs.
- One LIF handling management traffic must be configured for every storage virtual machine (SVM) supporting SAN.
- NVMe LIFs and namespaces must be hosted on the same node.
- Only one NVMe LIF handling data traffic can be configured per SVM

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 subnet_name} -firewall-policy data -auto -revert {true|false}`

Option	Description
ONTAP 9.5 and earlier	`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 subnet_name} -firewall -policy data -auto-revert {true|false}`
ONTAP 9.6 and later	`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 subnet_name} -firewall -policy data -auto-revert {true|false}`

   - The `-role` parameter is not required when creating a LIF using a service policy (beginning withONTAP 9.6).
   - 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.

     The `-data-protocol` parameter is not required when creating a LIF using a service policy (beginning with ONTAP 9.6).

   - `-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. The `network route create` man page contains information about creating a static route within an SVM.
- 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.

- `-auto-revert` allows you to specify whether a data LIF is automatically reverted to its home node under circumstances such as startup, changes to the status of the management database, or when the network connection is made. The default setting is `false`, but you can set it to `false` depending on network management policies in your environment.

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:

```
network interface create -vserver vs1.example.com -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):

```
network interface create -vserver vs3.example.com -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:

```
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.example.com
            datalif1     up/down  192.0.2.145/30   node-1       e1c
true
vs3.example.com
            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.
```

The following command shows how to create a NAS data LIF that is assigned with the `default-data-files` service policy:

```
network interface create -vserver vs1 -lif lif2 -home-node node2 -homeport
e0d -service-policy default-data-files -subnet-name ipspace1
```

Enable 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.

What you'll need

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

You should configure more than one DNS server to avoid a single-point-of-failure. The `vserver services name-service dns create` command issues a warning if you enter only one DNS server name.

About this task

The *Network Management Guide* contains information about configuring dynamic DNS on the SVM.

Steps

1. Enable DNS on the SVM:

   `vserver services name-service dns create -vserver` *`vserver_name`* `-domains` *`domain_name`* `-name-servers` *`ip_addresses`* `-state enabled`

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

   ```
   vserver services name-service dns create -vserver vs1.example.com
   -domains example.com -name-servers 192.0.2.201,192.0.2.202 -state
   enabled
   ```

   Beginning with ONTAP 9.2, the `vserver services name-service dns create` command performs an automatic configuration validation and reports an error message if ONTAP cannot contact the name server.

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

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

```
vserver services name-service dns show
                                                 Name
Vserver           State     Domains              Servers
----------------- --------- -------------------- -------------
cluster1          enabled   example.com          192.0.2.201,
                                                 192.0.2.202
vs1.example.com   enabled   example.com          192.0.2.201,
                                                 192.0.2.202
```

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

```
vserver services name-service dns show -vserver vs1.example.com
               Vserver: vs1.example.com
               Domains: example.com
         Name Servers: 192.0.2.201, 192.0.2.202
    Enable/Disable DNS: enabled
        Timeout (secs): 2
      Maximum Attempts: 1
```

3. Validate the status of the name servers by using the `vserver services name-service dns check` command.

   The `vserver services name-service dns check` command is available beginning with ONTAP 9.2.

```
vserver services name-service dns check -vserver vs1.example.com

Vserver           Name Server     Status       Status Details
---------------  -------------   ----------
--------------------------
vs1.example.com   10.0.0.50         up         Response time (msec): 2
vs1.example.com   10.0.0.51         up         Response time (msec): 2
```

Configure name services

Configure name services overview

Depending on the configuration of your storage system, 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 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.

Configure the name service switch table

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

What you'll need

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.

Configure local UNIX users and groups

Configure local UNIX users and groups overview

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.

Create 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"
```

Load 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
```

Create 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
```

Add 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
```

Load 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`, 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
```

Work with netgroups

Working with netgroups overview

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.

What you'll need

Before working with netgroups, you must ensure the following conditions are met:

- 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.

- The names of all hosts in netgroups, regardless of their source (NIS, LDAP, or local files), must be correctly spelled and use the correct case. Case inconsistencies in host names used in netgroups can lead to unexpected behavior, such as failed export checks.
- 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

When you work with netgroups, you can perform the following operations:

- 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` command 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.

Load 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`).

What you'll need

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.

  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 ONTAP does not support them.

  Only the host/IP part is supported.

About this task

ONTAP supports netgroup-by-host searches for the local netgroup file. After you load the netgroup file, 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 take 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
```

Verify 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
```

Create 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.

What you'll need

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, 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. Beginning with ONTAP 9.7, NIS `netgroup.byhost` entries can be cached using the `vserver services name-service nis-domain netgroup-database` commands.

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.

> Beginning with ONTAP 9.2, the field `-nis-servers` replaces the field `-servers`. This new field can take either a hostname or an IP address for the NIS server.

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 -nis-servers 192.0.2.180
```

Use LDAP

Overview of 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.

Beginning with ONTAP 9.10.1, LDAP channel binding is supported by default for both Active Directory and name services LDAP connections. ONTAP will try channel binding with LDAP connections only if Start-TLS or LDAPS is enabled along with session security set to either sign or seal. To disable or reenable LDAP channel binding with name servers, use the `-try-channel-binding` parameter with the `ldap client modify` command.

For more information, see 2020 LDAP channel binding and LDAP signing requirements for Windows.

- Before configuring LDAP for 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.
  - The LDAP user password hash types supported by the LDAP server must include those supported by ONTAP:
    - CRYPT (all types) and SHA-1 (SHA, SSHA).
    - Beginning with ONTAP 9.8, SHA-2 hashes (SHA-256, SSH-384, SHA-512, SSHA-256, SSHA-384, and SSHA-512) are also supported.
  - 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)
        - START TLS
        - LDAPS (LDAP over TLS or SSL)
    - 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 START TLS or LDAPS, the following points should be considered.
        - It is a NetApp best practice to use Start TLS rather than LDAPS.
        - If LDAPS is used, the LDAP server must be enabled for TLS or for SSL in ONTAP 9.5 and later. SSL is not supported in ONTAP 9.0-9.4.
        - A certificate server must already be configured in the domain.
    - To enable LDAP referral chasing (in ONTAP 9.5 and later), the following conditions must be satisfied:
        - Both domains should be configured with one of the following trust relationships:
            - Two-way
            - One-way, where the primary trusts the referral domain
            - Parent-child
        - DNS must be configured to resolve all referred server names.
        - Domain passwords should be same to authenticate when --bind-as-cifs-server set to true.

The following configurations are not supported with LDAP referral chasing.

- For all ONTAP versions:
    - LDAP clients on an admin SVM
- For ONTAP 9.8 and earlier (they are supported in 9.9.1 and later):
    - LDAP signing and sealing (the `-session-security` option)
    - Encrypted TLS connections (the `-use-start-tls` option)
    - Communications over LDAPS port 636 (the `-use-ldaps-for-ad-ldap` option)

- You must enter an LDAP schema when configuring the LDAP client on the SVM.

  In most cases, one of the default 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 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.

For additional information, see NetApp Technical Report 4835: How to Configure LDAP in ONTAP.

Create a new LDAP client schema

If the LDAP schema in your environment differs from the ONTAP defaults, you must

create a new LDAP client schema for ONTAP before creating the LDAP client configuration.

About this task

Most LDAP servers can use the default schemas provided by ONTAP:

- MS-AD-BIS (the preferred schema for most Windows 2012 and later AD servers)
- AD-IDMU (Windows 2008, Windows 2012 and later AD servers)
- AD-SFU (Windows 2003 and earlier AD servers)
- RFC-2307 (UNIX LDAP servers)

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

The default LDAP schemas provided by 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
   ```

Install 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 in ONTAP 9.0 and 9.1.

Beginning with ONTAP 9.2, all applications within ONTAP that use TLS communications can check digital certificate status using Online Certificate Status Protocol (OCSP). If OCSP is enabled for LDAP over TLS, revoked certificates are rejected and the connection fails.

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`

Create an LDAP client configuration

If you want ONTAP to access the external LDAP servers in your environment, you must first set up an LDAP client on the storage system.

What you'll need

One of the first three servers in the AD-domain resolved list must be up and serving data. Otherwise, this task fails.

There are multiple servers, out of which more than two servers are down at any point of time.

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 an 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 by 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.

The `-servers` option is deprecated in ONTAP 9.2. Beginning with ONTAP 9.2, the `-ldap-servers` field replaces the `-servers` field. This new field can take either a host name or an IP address for the LDAP server.

b. Specify a default or custom LDAP schema.

Most LDAP servers can use the default read-only schemas that are provided by 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 then modifying the copy.

Default schemas:

- MS-AD-BIS

  Based on RFC-2307bis, this is the preferred LDAP schema for most standard Windows 2012 and later LDAP deployments.

- `AD-IDMU`

  Based on Active Directory Identity Management for UNIX, this schema is appropriate for most Windows 2008, Windows 2012, and later AD servers.

- `AD-SFU`

  Based on Active Directory Services for UNIX, this schema 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*), this schema 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 value is **`anonymous`**.

- `-bind-dn` *`LDAP_DN`* specifies the bind user.

  For Active Directory servers, you must specify the user in the account (DOMAIN\user) or principal (user@domain.com) form. Otherwise, you must 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 value is **`false`**. You must install a self-signed root CA certificate of the LDAP server to use this option.

  > If the SVM has a SMB server added to a domain and the LDAP server is one of the domain controllers of the home-domain of the SMB 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 you must 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 1 through 10 seconds. The default value is `3` seconds.

- `-base-dn` *`LDAP_DN`* specifies the base DN.

  Multiple values can be entered if needed (for example, if LDAP referral chasing is enabled). The default value is `""` (root).

- `-base-scope {base|onelevel|subtree}` specifies the base search scope.

  The default value is `subtree`.

- `-referral-enabled {true|false}` specifies whether LDAP referral chasing is enabled.

  Beginning with ONTAP 9.5, this allows the ONTAP LDAP client to refer look-up requests to other LDAP servers if an LDAP referral response is returned by the primary LDAP server indicating that the desired records are present on referred LDAP servers. The default value is **`false`**.

  To search for records present in the referred LDAP servers, the base-dn of the referred records must be added to the base-dn as part of LDAP client configuration.

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} [-referral-enabled {true|false}]
```

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 creates a new LDAP client configuration named ldap1 for the SVM vs1 to work with an Active Directory server for LDAP where LDAP referral chasing 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=adbasedomain,DC=example1,DC=com; DC=adrefdomain,DC=example2,DC=com"
-base-scope subtree -preferred-ad-servers 172.17.32.100 -referral-enabled
true
```

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
```

The following command modifies the LDAP client configuration named ldap1 for the SVM vs1 by enabling referral chasing:

```
cluster1::> vserver services name-service ldap client modify -vserver vs1
-client-config ldap1 -base-dn "DC=adbasedomain,DC=example1,DC=com;
DC=adrefdomain,DC=example2,DC=com"  -referral-enabled true
```

Associate 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.

What you'll need

- 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.

Steps

1. Enable LDAP on the SVM:

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

   > Beginning with ONTAP 9.2, the `vserver services name-service ldap create` command performs an automatic configuration validation and reports an error message if ONTAP is unable to contact the name server.

   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
   ```

2. Validate the status of the name servers by using the vserver services name-service ldap check command.

   The following command validates LDAP servers on the SVM vs1.

```
cluster1::> vserver services name-service ldap check -vserver vs1

| Vserver: vs1                                                    |
| Client Configuration Name: c1                                   |
| LDAP Status: up                                                 |
| LDAP Status Details: Successfully connected to LDAP server
"10.11.12.13".                                                   |
```

The name service check command is available beginning with ONTAP 9.2.

Verify 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`

   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`

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

Use Kerberos with NFS for strong security

Overview of 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 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 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 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.

Verify 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.

Create an NFS Kerberos realm configuration

If you want 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.

What you'll need

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"
```

Configure NFS Kerberos permitted encryption types

By default, 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, 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 ONTAP, 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`
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`

Enable 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
   ```

   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 and 2 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.
```