Manage NFS trunking

ONTAP 9

NetApp
June 04, 2024

Table of Contents

Manage NFS trunking

NFS trunking overview

Beginning with ONTAP 9.14.1, NFSv4.1 clients can take advantage of session trunking to open multiple connections to different LIFs on the NFS server, thereby increasing the speed of data transfer and providing resiliency through multipathing.

Trunking is advantageous for exporting FlexVol volumes to trunking-capable clients, in particular VMware and Linux clients, or for NFS over RDMA, TCP, or pNFS.

In ONTAP 9.14.1, trunking is restricted to LIFs on a single node; trunking cannot span LIFs across multiple nodes.

FlexGroup volumes are supported for trunking. Although this can provide better performance, multipath access to a FlexGroup volume can only be configured on a single node.

Only session trunking is supported for multipathing in this release.

How to use trunking

To take advantage of multipathing benefits offered by trunking, you need a set of LIFs – referred to as a *trunking group* – that are associated with the SVM containing a trunking-enabled NFS server. The LIFs in a trunking group must have home ports on the same node of the cluster, and they must reside on those home ports. It is a best practice that all LIFs in a trunking group are members of the same failover group.

ONTAP supports up to 16 trunked connections per node from a given client.

When a client mounts exports from a trunking-enabled server, they specify a number of IP addresses for LIFs in a trunking group. After the client connects to the first LIF, additional LIFs are only added to the NFSv4.1 session and used for trunking if they conform to trunking group requirements. The client then distributes NFS operations over the multiple connections based on their own algorithm (such as round-robin).

For best performance, you should configure trunking in an SVM that is dedicated to providing multipath exports, not single-path exports. That is, you should only enable trunking on a NFS server in an SVM whose exports are provided to trunking-enabled clients only.

Supported clients

The ONTAP NFSv4.1 server supports trunking with any client capable of NFSv4.1 session trunking.

The following clients have been tested with ONTAP 9.14.1:

- VMware - ESXi 7.0U3F and later
- Linux - Red Hat Enterprise Linux (RHEL) 8.8 and 9.3

When trunking is enabled on an NFS server, users accessing exported shares on NFS clients that do not support trunking might see a performance drop. This is because only a single TCP connection is used for multiple mounts to the SVM data LIFs.

Difference between NFS trunking and nconnect

Beginning with ONTAP 9.8, nconnect functionality is available by default when NFSv4.1 is enabled. On nconnect-capable clients, a single NFS mount can have multiple TCP connections (up to 16) over a single LIF.

In contrast, trunking is *multipathing* functionality, which provides multiple TCP connections over multiple LIFs. If you have the ability to employ additional NICs in your environment, trunking provides increased parallelism and performance beyond the capability of nconnect.

Learn more about nconnect.

Configure a new NFS server and exports for trunking

Create a trunking-enabled NFS server

Beginning with ONTAP 9.14.1, trunking can be enabled on NFS servers. NFSv4.1 is enabled by default when NFS servers are created.

Before you begin

The SVM must be:

- backed by sufficient storage for client data requirements.
- enabled for NFS.
- dedicated to NFS trunking. No other clients should be configured on it.

Steps

1. If a suitable SVM does not exist, create one:

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

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

   ```
   vserver show -vserver svm_name
   ```

   Learn more about creating an SVM.

3. Create the NFS server:

   ```
   vserver nfs create -vserver svm_name -v3 disabled -v4.0 disabled -v4.1 enabled
   -v4.1-trunking enabled -v4-id-domain my_domain.com
   ```

4. Verify that NFS is running:

   ```
   vserver nfs status -vserver svm_name
   ```

5. Verify that NFS is configured as desired:

   ```
   vserver nfs show -vserver svm_name
   ```

   Learn more about NFS server configuration.

After you finish

Configure the following services as needed:

- DNS
- LDAP
- Kerberos

Prepare your network for trunking

To take advantage of NFSv4.1 trunking, the LIFs in a trunking group must reside on the same node and have home ports on the same node. The LIFs should be configured in a failover group on the same node.

About this task

A one-to-one mapping of LIFs and NICs yields the greatest performance gain but is not required to enable trunking. Having at least two NICs installed can offer a performance benefit, but it is not required.

You can have multiple failover groups, but the failover group for trunking should include only those LIFS in the trunking group.

You should adjust the trunking failover group any time you add or remove connections (and underlying NICs) from a failover group.

Before you begin

- You should know the port names associated with the NICs if you want to create a failover group.
- The ports must all be on the same node.

Steps

1. Verify the names and status of the network ports you plan to use:

   ```
   network port status
   ```

2. Create the failover group:

   ```
   network interface failover-groups create -vserver svm_name -failover-group
   failover_group_name -targets ports_list
   ```

   It is not a requirement to have a failover group, but it is strongly recommended.

   - *svm_name* is the name of the SVM containing the NFS server.
   - *ports_list* is the list of ports that will be added to the failover group.

     Ports are added in the format *node_name:port_number*, for example, node1:e0c.

     The following command creates failover group fg3 for SVM vs1 and adds three ports:

     ```
     network interface failover-groups create -vserver vs1 -failover-group fg3
     -targets cluster1-01:e0c,cluster1-01:e0d,cluster1-01:e0e
     ```

     Learn more about failover groups.

3. If needed, create LIFs for members of the trunking group:

   `network interface create -vserver` *`svm_name`* `-lif` *`lif_name`* `-home-node` *`node_name`* `-home-port` *`port_name`* `-address` *`IP_address`* `-netmask` *`IP_address`* `[-service-policy` *`policy`*`] [-auto-revert {true|false}]`

   - `-home-node` - 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, not with the `-subnet` option.
   - When you assign IP addresses, you may 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.
   - `-service-policy` - the service policy for the LIF. If no policy is specified, a default policy will be assigned automatically. Use the `network interface service-policy show` command to review available service policies.
   - `-auto-revert` - 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 true depending on network management policies in your environment.

     Repeat this step for every LIF in the trunking group.

     The following command creates `lif-A` for the SVM `vs1`, on port `e0c` of the node `cluster1_01`:

     `network interface create -vserver vs1 -lif lif-A -service-policy` **`???`** `-home-node cluster1_01 -home-port e0c -address 192.0.2.0`

     Learn more about LIF creation.

4. Verify the LIFs were created:

   `network interface show`

5. Verify the configured IP address is reachable:

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

Export data for client access

To provide client access to data shares, you must create one or more volumes, and the

volume must have export policies with at least one rule.

Client export requirements:

- Linux clients must have a separate mount and a separate mount point for each trunking connection (that is, for each LIF).
- VMware clients require only a single mount point for an exported volume, with multiple LIFs specified.

  VMware clients require root access in the export policy.

Steps

1. Create an export policy:

   `vserver export-policy create -vserver` *`svm_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
   ```

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

   `vserver export-policy rule create -vserver` *`svm_name`* `-policyname` *`policy_name`* `-ruleindex` *`integer`* `-protocol nfs4 -clientmatch {` *`text | "text,text,…"`* `} -rorule` *`security_type`* `-rwrule` *`security_type`* `-superuser` *`security_type`* `-anon` *`user_ID`*

   The `-clientmatch` parameter should identify the trunking-capable Linux or VMware clients that will mount the export.

   Learn more about creating export rules.

4. Create the volume with a junction point:

   `volume create -vserver` *`svm_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`*

   Learn about creating volumes.

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

   `volume show -vserver` *`svm_name`* `-volume` *`volume_name`* `-junction-path`

Create client mounts

Linux and VMware clients that support trunking can mount volumes or data shares from an ONTAP NFSv4.1 server that is enabled for trunking.

When entering mount commands on the clients, you must enter IP addresses for each LIF in the trunking group.

Learn about supported clients.

Linux client requirements

A separate mount point is required for each connection in the trunking group.

Mount the exported volumes with commands similar to the following:

```
mount lif1_ip:/vol-test /mnt/test1 -o vers=4.1,max_connect=16
```

```
mount lif2_ip:/vol-test /mnt/test2 -o vers=4.1,max_connect=16
```

The version (`vers`) value should be `4.1` or later.

The `max_connect` value corresponds to the number of connections in the trunking group.

VMware client requirements

A mount statement is required that includes an IP address for each connection in the trunking group.

Mount the exported datastore with a command similar to the following:

```
#esxcli storage nfs41 -H lif1_ip, lif2_ip -s /mnt/sh are1 -v nfs41share
```

The `-H` values correspond to the connections in the trunking group.

Adapt existing NFS exports for trunking

Adapting single-path exports overview

You can adapt an existing single-path (non-trunked) NFSv4.1 export to use trunking. Trunking-capable clients can take advantage of improved performance as soon as trunking is enabled on the server, provided the server and client prerequisites have been satisfied.

Adapting a single-path export for trunking allows you to maintain exported data sets in their existing volumes and SVMs. To do so, you must enable trunking on the NFS server, update networking and export configuration, and remount the exported share on the clients.

Enabling trunking has the effect of restarting the server. VMware clients must then remount the exported datastores; Linux clients must remount exported volumes with the `max_connect` option.

Enable trunking on the NFS server

Trunking must be explicitly enabled on NFS servers. NFSv4.1 is enabled by default when NFS servers are created.

After enabling trunking, verify that the following services are configured as needed.

- DNS
- LDAP
- Kerberos

Steps

1. Enable trunking and ensure that NFSv4.1 is enabled:

   `vserver nfs create -vserver svm_name -v4.1 enabled -v4.1-trunking enabled`

2. Verify that NFS is running: `vserver nfs status -vserver svm_name`
3. Verify that NFS is configured as desired:

   `vserver nfs show -vserver svm_name`

   Learn more about NFS server configuration. .. If you are serving to Windows clients from this SVM, move the shares then delete the server. `vserver cifs show -vserver svm_name`

   + `vserver cifs delete -vserver svm_name`

Update your network for trunking

NFSv4.1 trunking requires the LIFs in a trunking group to reside on the same node and have home ports on the same node. All LIFs should be configured in a failover group on the same node.

About this task

A one-to-one mapping of LIFs and NICs yields the greatest performance gain, but is not required to enable trunking.

You can have multiple failover groups, but the failover group for trunking must include only those LIFS in the trunking group.

You should adjust the trunking failover group any time you add or remove connections (and underlying NICs) from a failover group.

Before you begin

- You must know the port names associated with the NICs to create a failover group.
- The ports must all be on the same node.

Steps

1. Verify the names and status of the network ports you plan to use:

   `network port show`

2. Create a trunking failover group or modify an existing one for trunking:

   ```
   network interface failover-groups create -vserver svm_name -failover-group
   failover_group_name -targets ports_list
   ```

   ```
   network interface failover-groups modify -vserver svm_name -failover-group
   failover_group_name -targets ports_list
   ```

   It is not a requirement to have a failover group, but it is strongly recommended.

   - *svm_name* is the name of the SVM containing the NFS server.
   - *ports_list* is the list of ports that will be added to the failover group.

     Ports are added in the format *node_name:port_number*, for example, `node1:e0c`.

     The following command creates failover group `fg3` for SVM vs1 and adds three ports:

     ```
     network interface failover-groups create -vserver vs1 -failover-group fg3
     -targets cluster1-01:e0c,cluster1-01:e0d,cluster1-01:e0e
     ```

     Learn more about failover groups.

3. Create additional LIFs for members of the trunking group as needed:

   ```
   network interface create -vserver svm_name -lif lif_name -home-node node_name
   -home-port port_name -address IP_address -netmask IP_address [-service-policy
   policy] [-auto-revert {true|false}]
   ```

   - `-home-node` - the node to which the LIF returns when the network interface revert command is run on the LIF.

     You can 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.
   - When 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.
   - `-service-policy` - the service policy for the LIF. If no policy is specified, a default policy will be assigned automatically. Use the `network interface service-policy show` command to review available service policies.
   - `-auto-revert` - 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 true depending on network management policies in your environment.

     Repeat this step for each additional LIF needed in the trunking group.

The following command creates lif-A for the SVM vs1, on port e0c of the node cluster1_01:

```
network interface create -vserver vs1 -lif lif-A -service-policy default-
intercluster -home-node cluster1_01 -home-port e0c -address 192.0.2.0
```

Learn more about LIF creation.

4. Verify that the LIFs were created:

```
network interface show
```

5. Verify that the configured IP address is reachable:

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

Modify data export for client access

To enable clients to take advantage of trunking for existing data shares, you might have to modify export policies and rules, and the volumes to which they are attached. There are different export requirements for Linux clients and VMware datastores.

Client export requirements:

- Linux clients must have a separate mount and a separate mount point for each trunking connection (that is, for each LIF).

  If you are upgrading to ONTAP 9.14.1 and you have already exported a volume, you can continue to use that volume in a trunking group.

- VMware clients require only a single mount point for an exported volume, with multiple LIFs specified.

  VMware clients require root access in the export policy.

Steps

1. Verify that an existing export policy is in place:

```
vserver export-policy show
```

2. Verify that the existing export policy rules are appropriate for the trunking configuration:

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

In particular, verify that the `-clientmatch` parameter correctly identifies the trunking-capable Linux or VMware clients that will mount the export.

If adjustments are necessary, modify the rule using the `vserver export-policy rule modify` command or create a new rule:

```
vserver export-policy rule create -vserver svm_name -policyname policy_name
```

```
-ruleindex integer -protocol nfs4 -clientmatch { text | "text,text,…" }
-rorule security_type -rwrule security_type -superuser security_type -anon
user_ID
```

Learn more about creating export rules.

3. Verify that existing exported volumes are online:

```
volume show -vserver svm_name
```

Reestablish client mounts

To convert non-trunked client connections to trunked connections, existing mounts on Linux and VMware clients must be unmounted and remounted using information about LIFs.

When entering mount commands on the clients, you must enter IP addresses for each LIF in the trunking group.

Learn about supported clients.

> Unmounting VMware clients is disruptive for any VMs on the datastore. An alternative would be to create a new datastore enabled for trunking, and use **storage vmotion** to move your VMs from the old datastore to the new one. See your VMware documentation for details.

Linux client requirements

A separate mount point is required for each connection in the trunking group.

Mount the exported volumes with commands similar to the following:

```
mount lif1_ip:/vol-test /mnt/test1 -o vers=4.1,max_connect=2

mount lif2_ip:/vol-test /mnt/test2 -o vers=4.1,max_connect=2
```

The `vers` value should should be `4.1` or later.

The `max_connect` value should correspond to the number of connections in the trunking group.

VMware client requirements

A mount statement is required that includes an IP address for each connection in the trunking group.

Mount the exported datastore with a command similar to the following:

```
#esxcli storage nfs41 -H lif1_ip, lif2_ip -s /mnt/sh are1 -v nfs41share
```

The `-H` values should correspond to the connections in the trunking group.

Copyright information

Trademark information