## Contents

Deciding whether to use this guide ............................................................. 7  
Networking components of a cluster .......................................................... 8  
Workflow: NAS path failover ................................................................. 10  
  Worksheet for NAS path failover configuration ........................................... 11  
  Creating an IPspace ................................................................................. 17  
  Determining which ports can be used for a broadcast domain ....................... 17  
  Removing ports from an existing broadcast domain ........................................ 19  
  Creating a broadcast domain ....................................................................... 20  
  Creating a subnet ....................................................................................... 21  
  Creating an SVM ....................................................................................... 22  
  Configuring LIFs on the SVM ..................................................................... 24  
  Configuring DNS services for the SVM ....................................................... 25  
  Configuring dynamic DNS on the SVM ....................................................... 27  
Configuring network ports ........................................................................ 28  
  Types of network ports ............................................................................. 28  
  Combining physical ports to create interface groups ....................................... 28  
    Interface group types ............................................................................. 29  
    Creating an interface group ..................................................................... 32  
    Adding a port to an interface group ........................................................... 33  
    Removing a port from an interface group .................................................. 33  
    Deleting an interface group ..................................................................... 34  
  Configuring VLANs over physical ports ...................................................... 34  
    Creating a VLAN ................................................................................... 35  
    Deleting a VLAN ................................................................................... 36  
  Modifying network port attributes ............................................................... 36  
  Modifying MTU setting for interface group ports ......................................... 37  
  Monitoring the health of network ports ......................................................... 38  
  Converting 40GbE NIC ports into multiple 10GbE ports for 10GbE connectivity ................................................................. 39  
  Removing a NIC from the node ................................................................... 40  
Configuring IPspaces ............................................................................... 42  
  Example of using IPspaces ........................................................................ 42  
  Standard properties of IPspaces .................................................................. 44  
  Creating IPspaces ..................................................................................... 45  
  Displaying IPspaces .................................................................................. 46  
  Deleting an IPspace .................................................................................. 46  
Configuring broadcast domains ............................................................... 48  
  Example of using broadcast domains .......................................................... 48  
  Creating a broadcast domain ....................................................................... 49  
  Adding or removing ports from a broadcast domain ....................................... 50  
  Splitting broadcast domains ....................................................................... 52
Merging broadcast domains ................................................................. 52
Changing the MTU value for ports in a broadcast domain .................. 53
Displaying broadcast domains ............................................................. 54
Deleting a broadcast domain ............................................................... 54

**Configuring failover groups and policies for LIFs** ............................... 56
Creating a failover group .................................................................... 56
Configuring failover settings on a LIF .............................................. 57
Commands for managing failover groups and policies ....................... 59

**Configuring subnets** ........................................................................ 60
Creating a subnet .................................................................................. 60
Adding or removing IP addresses from a subnet ................................. 61
Changing subnet properties ................................................................. 62
Displaying subnets .............................................................................. 63
Deleting a subnet .................................................................................. 64

**Configuring LIFs** ............................................................................. 65
What LIFs are ....................................................................................... 65
Roles for LIFs ....................................................................................... 66
Characteristics of LIFs ........................................................................ 67
Configuring LIF service policies ........................................................ 69
Creating a service policy for LIFs ....................................................... 69
Assigning a service policy to a LIF ..................................................... 71
Commands for managing LIF service policies ................................... 72
Creating a LIF ..................................................................................... 72
Modifying a LIF .................................................................................. 75
Migrating a LIF .................................................................................... 76
Reverting a LIF to its home port ......................................................... 78
Deleting a LIF ..................................................................................... 78
Configuring virtual IP (VIP) LIFs .......................................................... 79
Setting up Border Gateway Protocol (BGP) ...................................... 79
Creating a virtual IP (VIP) data LIF ................................................... 81
Commands for managing the Border Gateway Protocol (BGP) .......... 82

**Configuring host-name resolution** .................................................. 83
Configuring DNS for host-name resolution ........................................ 83
Configuring an SVM and data LIFs for host-name resolution using an
external DNS server ............................................................................ 83
Configuring the name service switch table for host-name resolution ... 84
Managing the hosts table ..................................................................... 85
Commands for managing local host-name entries .............................. 85

**Balancing network loads to optimize user traffic** ............................... 86
What DNS load balancing is ............................................................... 86
How DNS load balancing works ......................................................... 86
Creating a DNS load balancing zone .................................................. 86
Adding or removing a LIF from a load balancing zone ...................... 87

**Configuring network security using Federal Information Processing
Standards (FIPS)** ............................................................................. 89
Enabling FIPS ................................................................. 89
Disabling FIPS ............................................................. 90
Viewing FIPS compliance status ................................... 90

**Configuring IPv6 addresses** ........................................ 92

- Enabling IPv6 on the cluster ........................................ 92
- Enabling or disabling RA processing ............................ 93

**Configuring QoS marking** ........................................... 94

- DSCP marking for UC Compliance ............................... 94
- Modifying QoS marking values .................................... 94
- Displaying QoS marking values .................................... 95

**Configuring firewall service and policies for LIFs** .......... 96

- LIF roles and default firewall policies ......................... 96
- Portmap service is configurable in firewall in ONTAP 9.4 .... 97
- Creating a firewall policy and assigning it to a LIF ........ 98
- Commands for managing firewall service and policies .... 100

**Managing routing in an SVM** ...................................... 101

- Creating a static route ............................................... 101
- Enabling multipath routing ......................................... 101
- Deleting a static route ............................................... 102
- Displaying routing information .................................... 102
- Removing dynamic routes from routing tables ............... 104

**Managing SNMP on the cluster** ................................... 105

- What MIBs are ......................................................... 105
- SNMP traps .............................................................. 106
- Creating an SNMP community and assigning it to a LIF .... 106
- Configuring SNMPv3 users in a cluster ....................... 108
  - SNMPv3 security parameters ................................. 109
  - Examples for different security levels .................... 109
- Configuring traphosts to receive SNMP notifications .... 111
- Commands for managing SNMP ................................. 112

**ONTAP port usage on a storage system** ....................... 114

**Viewing network information** ..................................... 118

- Displaying network port information ......................... 118
- Displaying information about a VLAN ....................... 119
- Displaying interface group information ..................... 120
- Displaying LIF information ....................................... 121
- Displaying routing information .................................. 122
- Displaying host name entries .................................... 124
- Displaying DNS domain configurations ..................... 124
- Displaying information about failover groups .............. 125
- Displaying LIF failover targets .................................. 126
- Displaying LIFs in a load balancing zone .................... 127
- Displaying cluster connections ................................. 128
  - Displaying active connections by client .................. 128
  - Displaying active connections by protocol ............... 129
Deciding whether to use the Network Management Guide

This guide describes basic storage network administration. It shows you how to configure physical and virtual network ports (VLANs and interface groups), how to create LIFs using IPv4 and IPv6, how to manage routing and host-resolution services in clusters, how to use load balancing to optimize network traffic, and how to monitor a cluster using SNMP.

**You should use this guide** under the following circumstances:

- You want to understand the range of ONTAP network management capabilities.
- You want to use the command-line interface (CLI), not OnCommand System Manager.

**If you want to use OnCommand System Manager** to configure the network, you should choose the following documentation:

- *Cluster management using System Manager*

**If you require additional configuration or conceptual information**, you should choose among the following documentation:

- Conceptual background for network configuration
  - *ONTAP concepts*
- NAS file access
  - *NFS management*
  - *SMB/CIFS management*
- SAN host provisioning
  - *SAN administration*
- Command reference
  - *ONTAP 9 commands*
- Technical Reports (TRs), which include additional information about ONTAP technology and interaction with external services
Networking components of a cluster

You should familiarize yourself with the networking components of a cluster before setting up the cluster. Configuring the physical networking components of a cluster into logical components provides the flexibility and multi-tenancy functionality in ONTAP.

The various networking components in a cluster are as follows:

- **Physical ports**
  Network interface cards (NICs) and host bus adapters (HBAs) provide physical (Ethernet and Fibre Channel) connections from each node to the physical networks (management and data networks).
  Default roles were assigned to each Ethernet network port. The roles included data, cluster, cluster management, intercluster, and node management. In ONTAP 9, these roles are assigned when LIFs are created.
  For site requirements, switch information, port cabling information, and controller onboard port cabling, see the *Hardware Universe* at hwu.netapp.com.

- **Logical ports**
  Virtual local area networks (VLANs) and interface groups (ifgrps) constitute the logical ports. Although interface groups treat several physical ports as a single port, VLANs subdivide a physical port into multiple separate ports.

- **IPspaces**
  You can use an *IPspace* to create a distinct IP address space for each SVM in a cluster. Doing so enables clients in administratively separate network domains to access cluster data while using overlapping IP addresses from the same IP address subnet range.

- **Broadcast domains**
  A broadcast domain resides in an IPspace and contains a group of network ports, potentially from many nodes in the cluster, that belong to the same layer 2 network. The ports in the group are used in an SVM for data traffic.

- **Subnets**
  A subnet is created within a broadcast domain and contains a pool of IP addresses that belong to the same layer 3 subnet. Addresses in a subnet are allocated to ports in the broadcast domain when a LIF is created.

- **Logical interfaces**
  A logical interface (LIF) is an IP address or a worldwide port name (WWPN) that is associated with a port. It is associated with attributes such as failover groups, failover rules, and firewall rules. A LIF communicates over the network through the port (physical or logical) to which it is currently bound.
  The different types of LIFs in a cluster are data LIFs, cluster management LIFs, node management LIFs, intercluster LIFs, and cluster LIFs. The ownership of the LIFs depends on the SVM where the LIF resides. Data LIFs are owned by data SVMs, node management LIFs, cluster management, and intercluster LIFs are owned by the admin SVMs, and cluster LIFs are owned by the cluster SVM.

- **DNS zones**
  DNS zone can be specified during the LIF creation, providing a name for the LIF to be exported through the cluster's DNS server. Multiple LIFs can share the same name, allowing the DNS load balancing feature to distribute IP addresses for the name according to load. SVMs can have multiple DNS zones.

- **Routing**
Each SVM is self sufficient with respect to networking. An SVM owns LIFs and routes that can reach each of the configured external servers.

The following figure illustrates how the different networking components are associated in a four-node cluster:

![Networking components of a cluster](image)

**Related information**

*ONTAP concepts*
Workflow: NAS path failover

If you are already familiar with basic networking concepts, you may be able to save time setting up your network by reviewing this “hands on” workflow for NAS path failover configuration.

A NAS LIF automatically migrates to a surviving network port after a link failure on its current port. If your network is flat, you can rely on the ONTAP defaults to manage path failover. Otherwise, you should configure path failover following the steps in this workflow.

**Note:** A SAN LIF does not migrate (unless you move it manually after the link failure). Instead, multipathing technology on the host diverts traffic to a different LIF. For more information, see the *SAN Administration Guide*.

---

**Related information**

*SAN administration*
Worksheet for NAS path failover configuration

You should complete all sections of the worksheet before configuring NAS path failover.

**IPspace configuration**

You can use an *IPspace* to create a distinct IP address space for each SVM in a cluster. Doing so enables clients in administratively separate network domains to access cluster data while using overlapping IP addresses from the same IP address subnet range.

<table>
<thead>
<tr>
<th>Information</th>
<th>Required?</th>
<th>Your values</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>IPspace name</em></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>• The name of the IPspace.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• The name must be unique in the cluster.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Broadcast domain configuration**

A broadcast domain groups ports that belong in the same Layer 2 network, and sets the MTU for the broadcast domain ports.

Broadcast domains are assigned to an IPspace. An IP space can contain one or more broadcast domains.

*Note:* The port to which a LIF fails over must be a member of the failover group for the LIF. When you create a broadcast domain, ONTAP automatically creates a failover group with the same name. The failover group contains all the ports assigned to the broadcast domain.

<table>
<thead>
<tr>
<th>Information</th>
<th>Required?</th>
<th>Your values</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>IPspace name</em></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>• The IPspace to which the broadcast domain is assigned.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• The IPspace must exist.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><em>Broadcast domain name</em></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>• The name of the broadcast domain.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• The name must be unique in the IPspace.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><em>MTU</em></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>• The MTU of the broadcast domain.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• You can specify either <strong>1500</strong> or <strong>9000</strong>.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• The MTU value is applied to all ports in the broadcast domain and to any ports that are later added to the broadcast domain.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Note:* The MTU value must match all the devices connected to that network.
### Ports
- The network ports to add to the broadcast domain.
- The ports assigned to the broadcast domain can be physical ports, VLANs, or interface groups (ifgroups).
- If a port is in another broadcast domain, it must be removed before it can be added to the broadcast domain.
- Ports are assigned by specifying both the node name and port: for example, “node1:e0d”.

### Subnet configuration
A subnet contains pools of IP addresses and a default gateway that can be assigned to LIFs used by SVMs residing in the IPspace.
- When creating a LIF on an SVM, you can specify the name of the subnet instead of supplying an IP address and a subnet.
- Since a subnet can be configured with a default gateway, you do not have to create the default gateway in a separate step when creating an SVM.
- A broadcast domain can contain one or more subnets. You can configure SVM LIFs that are on different subnets by associating more than one subnet with the IPspace's broadcast domain.
- Each subnet must contain IP addresses that do not overlap with IP addresses assigned to other subnets in the same IPspace.
- You can assign specific IP addresses to SVM data LIFs and create a default gateway for the SVM instead of using a subnet.

<table>
<thead>
<tr>
<th>Information</th>
<th>Required?</th>
<th>Your values</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>IPspace name</strong></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>• The IPspace to which the subnet will be assigned.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• The IPspace must exist.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Subnet name</strong></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>• The name of the subnet.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• The name must be unique in the IPspace.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Broadcast domain name</strong></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>• The broadcast domain to which the subnet will be assigned.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• The broadcast domain must reside in the specified IPspace.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Subnet name and mask</strong></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>• The subnet address and mask in which the IP addresses reside.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Gateway</strong></td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>• You can specify a default gateway for the subnet.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• If you do not assign a gateway when you create the subnet, you can assign one to the subnet at any time.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Information Required?

<table>
<thead>
<tr>
<th><strong>IP address ranges</strong></th>
<th>Required?</th>
<th>Your values</th>
</tr>
</thead>
<tbody>
<tr>
<td>• You can specify a range of IP addresses or specific IP addresses. For example, you can specify a range such as “192.168.1.1-192.168.1.100, 192.168.1.112, 192.168.1.145”. • If you do not specify an IP address range, the entire range of IP addresses in the specified subnet are available to assign to LIFs.</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Force update of LIF associations</strong></th>
<th>Required?</th>
<th>Your values</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Specifies whether to force the update of existing LIF associations. • By default, subnet creation fails if any service processor interfaces or network interfaces are using the IP addresses in the ranges provided. • Using this parameter associates any manually addressed interfaces with the subnet and allows the command to succeed.</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>

### SVM configuration

You use SVMs to serve data to clients and hosts.

The values you record are for creating a default data SVM. If you are creating a MetroCluster source SVM, see the Fabric-attached MetroCluster Installation and Configuration Guide or the Stretch MetroCluster Installation and Configuration Guide.

<table>
<thead>
<tr>
<th><strong>Information</strong></th>
<th>Required?</th>
<th>Your values</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SVM name</strong></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>• The name of the SVM. • You should use a fully qualified domain name (FQDN) to ensure unique SVM names across cluster leagues.</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td><strong>Root volume name</strong></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>• The name of the SVM root volume.</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td><strong>Aggregate name</strong></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>• The name of the aggregate that holds the SVM root volume. • The aggregate must exist.</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td><strong>Security style</strong></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>• The security style for the SVM root volume. • Possible values are ntfs, unix, and mixed.</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td><strong>IPspace name</strong></td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>• The IPspace to which the SVM is assigned. • The IPspace must exist.</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>
### SVM language setting

- The default language to use for the SVM and its volumes.
- If you do not specify a default language, the default SVM language is set to C.UTF-8.
- The SVM language setting determines the character set used to display file names and data for all NAS volumes in the SVM.

The language of the SVM can be modified after the SVM is created.

<table>
<thead>
<tr>
<th>Information</th>
<th>Required?</th>
<th>Your values</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM language setting</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>

### LIF configuration

An SVM serves data to clients and hosts through one or more network logical interfaces (LIFs).

<table>
<thead>
<tr>
<th>Information</th>
<th>Required?</th>
<th>Your values</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM name</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>LIF name</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>LIF role</td>
<td>Yes</td>
<td>data</td>
</tr>
<tr>
<td>Allowed protocols</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>

**Important:** If you are configuring a CIFS server to host Hyper-V or SQL Server over SMB for nondisruptive operation solutions, the SVM must have at least one data LIF on every node in the cluster.

**Note:** The protocols that use the LIF cannot be modified after the LIF is created. You should specify all protocols when you configure the LIF.
## Service policy

Service policy for the LIF (available starting with ONTAP 9.5)
The service policy defines which network services can use the LIF. Starting with ONTAP 9.5, a small number of network services can be included in a LIF’s service policy.

<table>
<thead>
<tr>
<th>Information</th>
<th>Required?</th>
<th>Your values</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Service policy</strong></td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>

## Home node

- The node to which the LIF returns when the LIF is reverted to its home port.
- You should record a home node for each data LIF.

<table>
<thead>
<tr>
<th>Information</th>
<th>Required?</th>
<th>Your values</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Home node</strong></td>
<td>Yes</td>
<td></td>
</tr>
</tbody>
</table>

## Home port

- The port to which the logical interface returns when the LIF is reverted to its home port.
- You should record a home port for each data LIF.

<table>
<thead>
<tr>
<th>Information</th>
<th>Required?</th>
<th>Your values</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Home port</strong></td>
<td>Yes</td>
<td></td>
</tr>
</tbody>
</table>

## Subnet name

- The subnet to assign to the SVM.
- All data LiFs used to create continuously available SMB connections to application servers must be on the same subnet.

<table>
<thead>
<tr>
<th>Information</th>
<th>Required?</th>
<th>Your values</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Subnet name</strong></td>
<td>Yes (if using a subnet)</td>
<td></td>
</tr>
</tbody>
</table>

## DNS configuration

You must configure DNS on the SVM before creating an NFS or CIFS server.

<table>
<thead>
<tr>
<th>Information</th>
<th>Required?</th>
<th>Your values</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DNS domain name</strong></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td><strong>SVM name</strong></td>
<td>Yes</td>
<td></td>
</tr>
</tbody>
</table>
### Information

**IP addresses of the DNS servers**
- List of IP addresses for the DNS servers that will provide name resolution for the NFS or CIFS server.
- The listed DNS servers must contain the service location records (SRV) needed to locate the Active Directory LDAP servers and domain controllers for the domain that the CIFS server will join.

The SRV record is used to map the name of a service to the DNS computer name of a server that offers that service. CIFS server creation fails if ONTAP cannot obtain the service location records through local DNS queries.

The simplest way to ensure that ONTAP can locate the Active Directory SRV records is to configure Active Directory-integrated DNS servers as the SVM DNS servers.

You can use non-Active Directory-integrated DNS servers provided that the DNS administrator has manually added the SRV records to the DNS zone that contains information about the Active Directory domain controllers.

- For information about the Active Directory-integrated SRV records, see the topic *How DNS Support for Active Directory Works* on Microsoft TechNet.

---

**Dynamic DNS configuration**

Before you can use dynamic DNS to automatically add DNS entries to your Active Directory-integrated DNS servers, you must configure dynamic DNS (DDNS) on the SVM.

DNS records are created for every data LIF on the SVM. By creating multiple data LIFS on the SVM, you can load-balance client connections to the assigned data IP addresses. DNS load balances connections that are made using the host name to the assigned IP addresses in a round-robin fashion.

<table>
<thead>
<tr>
<th>Information</th>
<th>Required?</th>
<th>Your values</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SVM name</strong></td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>• The SVM on which you want to create an NFS or CIFS server.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>What to use DDNS</th>
<th>Yes</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>• Specifies whether to use DDNS.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• The DNS servers configured on the SVM must support DDNS. By default, DDNS is disabled.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>What to use secure DDNS</th>
<th>No</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>• Secure DDNS is supported only with Active Directory-integrated DNS.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• If your Active Directory-integrated DNS allows only secure DDNS updates, the value for this parameter must be true.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• By default, secure DDNS is disabled.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Secure DDNS can be enabled only after a CIFS server or an Active Directory account has been created for the SVM.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Information

<table>
<thead>
<tr>
<th>Information</th>
<th>Required?</th>
<th>Your values</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FQDN of the DNS domain</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• The FQDN of the DNS domain.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• You must use the same domain name configured for DNS name services on the SVM.</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>

Related information

*Stretch MetroCluster installation and configuration*

*Fabric-attached MetroCluster installation and configuration*

Creating an IPspace

You can use an IPspace to create a distinct IP address space for each SVM in a cluster. Doing so enables clients in administratively separate network domains to access cluster data while using overlapping IP addresses from the same IP address subnet range.

Before you begin

You must be a cluster administrator to perform this task.

Step

1. Create an IPspace.

Example

```bash
network ipspace create -ipspace ipspace1
```

```bash
network ipspace show
```

<table>
<thead>
<tr>
<th>IPspace</th>
<th>Vserver List</th>
<th>Broadcast Domains</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cluster</td>
<td>Cluster</td>
<td>Cluster</td>
</tr>
<tr>
<td>Default</td>
<td>cluster1</td>
<td>Default</td>
</tr>
<tr>
<td>ipspace1</td>
<td>ipspace1</td>
<td>-</td>
</tr>
</tbody>
</table>

The IPspace is created, along with the system SVM for the IPspace. The system SVM carries management traffic.

Determining which ports can be used for a broadcast domain

Before you can configure a broadcast domain to add to the new IPspace, you must determine what ports are available for the broadcast domain.

Before you begin

You must be a cluster administrator to perform this task.

About this task

• Ports can be physical ports, VLANs, or interface groups (ifgroups).
• The ports that you want to add to the new broadcast domain cannot be assigned to an existing broadcast domain.
• If the ports that you want to add to the broadcast domain are already in another broadcast domain (for example, the Default broadcast domain in the Default IPspace), you must remove the ports from that broadcast domain before assigning them to the new broadcast domain.
• Ports that have LIFs assigned to them cannot be removed from a broadcast domain.
• Because the cluster management and node management LIFs are assigned to the Default broadcast domain in the Default IPspace, the ports assigned to these LIFs cannot be removed from the Default broadcast domain.

Steps
1. Determine the current port assignments.

Example

```
network port show
```

<table>
<thead>
<tr>
<th>Node</th>
<th>Port</th>
<th>IPspace</th>
<th>Broadcast Domain</th>
<th>Link</th>
<th>MTU</th>
<th>Admin/Oper</th>
</tr>
</thead>
<tbody>
<tr>
<td>node1</td>
<td>e0a</td>
<td>Cluster</td>
<td>Cluster</td>
<td>up</td>
<td>9000</td>
<td>auto/1000</td>
</tr>
<tr>
<td></td>
<td>e0b</td>
<td>Cluster</td>
<td>Cluster</td>
<td>up</td>
<td>9000</td>
<td>auto/1000</td>
</tr>
<tr>
<td></td>
<td>e0c</td>
<td>Default</td>
<td>Default</td>
<td>up</td>
<td>1500</td>
<td>auto/1000</td>
</tr>
<tr>
<td></td>
<td>e0d</td>
<td>Default</td>
<td>Default</td>
<td>up</td>
<td>1500</td>
<td>auto/1000</td>
</tr>
<tr>
<td></td>
<td>e0e</td>
<td>Default</td>
<td>Default</td>
<td>up</td>
<td>1500</td>
<td>auto/1000</td>
</tr>
<tr>
<td></td>
<td>e0f</td>
<td>Default</td>
<td>Default</td>
<td>up</td>
<td>1500</td>
<td>auto/1000</td>
</tr>
<tr>
<td></td>
<td>e0g</td>
<td>Default</td>
<td>Default</td>
<td>up</td>
<td>1500</td>
<td>auto/1000</td>
</tr>
<tr>
<td>node2</td>
<td>e0a</td>
<td>Cluster</td>
<td>Cluster</td>
<td>up</td>
<td>9000</td>
<td>auto/1000</td>
</tr>
<tr>
<td></td>
<td>e0b</td>
<td>Cluster</td>
<td>Cluster</td>
<td>up</td>
<td>9000</td>
<td>auto/1000</td>
</tr>
<tr>
<td></td>
<td>e0c</td>
<td>Default</td>
<td>Default</td>
<td>up</td>
<td>1500</td>
<td>auto/1000</td>
</tr>
<tr>
<td></td>
<td>e0d</td>
<td>Default</td>
<td>Default</td>
<td>up</td>
<td>1500</td>
<td>auto/1000</td>
</tr>
<tr>
<td></td>
<td>e0e</td>
<td>Default</td>
<td>Default</td>
<td>up</td>
<td>1500</td>
<td>auto/1000</td>
</tr>
<tr>
<td></td>
<td>e0f</td>
<td>Default</td>
<td>Default</td>
<td>up</td>
<td>1500</td>
<td>auto/1000</td>
</tr>
<tr>
<td></td>
<td>e0g</td>
<td>Default</td>
<td>Default</td>
<td>up</td>
<td>1500</td>
<td>auto/1000</td>
</tr>
</tbody>
</table>

In this example, the output from the command provides the following information:
• Ports e0c, e0d, e0e, e0f, and e0g on each node are assigned to the Default broadcast domain.
• These ports are potentially available to use in the broadcast domain of the IPspace that you want to create.

2. Determine which ports in the Default broadcast domain are assigned to LIF interfaces, and therefore cannot be moved to a new broadcast domain.

Example

```
network interface show
```

<table>
<thead>
<tr>
<th>Vserver</th>
<th>Logical Interface</th>
<th>Status Admin/Oper</th>
<th>Network Address/Mask</th>
<th>Current Node</th>
<th>Current Port</th>
<th>Current Is</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cluster</td>
<td>node1_clus1</td>
<td>up/up</td>
<td>10.0.2.40/24</td>
<td>node1</td>
<td>e0a</td>
<td>true</td>
</tr>
<tr>
<td></td>
<td>node1_clus2</td>
<td>up/up</td>
<td>10.0.2.41/24</td>
<td>node1</td>
<td>e0b</td>
<td>true</td>
</tr>
<tr>
<td></td>
<td>node2_clus1</td>
<td>up/up</td>
<td>10.0.2.42/24</td>
<td>node2</td>
<td>e0a</td>
<td>true</td>
</tr>
<tr>
<td></td>
<td>node2_clus2</td>
<td>up/up</td>
<td>10.0.2.43/24</td>
<td>node2</td>
<td>e0b</td>
<td>true</td>
</tr>
<tr>
<td></td>
<td>cluster_mgmt</td>
<td>up/up</td>
<td>10.0.1.41/24</td>
<td>node1</td>
<td>e0c</td>
<td>true</td>
</tr>
</tbody>
</table>
In the following example, the output from the command provides the following information:

- The node ports are assigned to port `e0c` on each node and the cluster administrative LIF’s home node is on `e0c` on node1.
- Ports `e0d, e0e, e0f, and e0g` on each node are not hosting LIFs and can be removed from the Default broadcast domain and then added to a new broadcast domain for the new IPspace.

### Removing ports from an existing broadcast domain

If the ports that you want to add to the new broadcast domain are already in another broadcast domain, you must remove the ports from that broadcast domain before assigning them to the new broadcast domain.

**Before you begin**

You must be a cluster administrator to perform this task.

**Steps**

1. Remove ports from the Default broadcast domain.

   **Example**

   ```
   network port broadcast-domain remove-ports -ipspace Default -broadcast-domain Default -ports node1:e0d,node1:e0e,node2:e0d,node2:e0e
   ```

2. Verify that the ports are not assigned to a broadcast domain.

   **Example**

   ```
   network port show
   ```
Creating a broadcast domain

You must create a broadcast domain for a custom IPspace. The SVMs created in the IPspace use the ports in the broadcast domain.

Before you begin

You must be a cluster administrator to perform this task.

About this task

The port to which a LIF fails over must be a member of the failover group for the LIF. When you create a broadcast domain, ONTAP automatically creates a failover group with the same name. The failover group contains all the ports assigned to the broadcast domain.

Steps

1. Create a broadcast domain.

   Example

   ```
   network port broadcast-domain create -ipspace ipspace1 -broadcast-domain -ipspace1 -mtu 1500 -ports node1:e0d,node1:e0e,node2:e0d,node2:e0e
   ```

2. Verify that the broadcast domain configuration is correct.

   a. Verify the broadcast domain is correct: `network port broadcast-domain show`

   b. Verify the network port is correct: `network port show`

   c. Verify the failover group names and failover targets are correct: `network interface failover-groups show`

   Example

   ```
   network port broadcast-domain show
   ```

<table>
<thead>
<tr>
<th>IPspace</th>
<th>Broadcast Domain Name</th>
<th>MTU</th>
<th>Port List</th>
<th>Update Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cluster</td>
<td>Cluster</td>
<td>9000</td>
<td>node1:e0a, node1:e0b, node2:e0a, node2:e0b</td>
<td>complete</td>
</tr>
<tr>
<td>Default</td>
<td>Default</td>
<td>1500</td>
<td>node1:e0c, node1:e0f, node1:e0g, node2:e0c, node2:e0f, node2:e0g</td>
<td>complete</td>
</tr>
<tr>
<td>ipspace1</td>
<td>ipspace1</td>
<td>1500</td>
<td>node1:e0d, node1:e0e, node2:e0d, node2:e0e</td>
<td>complete</td>
</tr>
</tbody>
</table>

   ```
   network port show
   ```

   20 Network Management Guide
network interface failover-groups show

<table>
<thead>
<tr>
<th>Vserver</th>
<th>Group</th>
<th>Failover Targets</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cluster</td>
<td>Cluster</td>
<td>node1:e0a, node1:e0b, node2:e0a, node2:e0b</td>
</tr>
<tr>
<td>cluster1</td>
<td>Default</td>
<td>node1:e0c, node1:e0f, node2:e0c, node2:e0f, node2:e0e</td>
</tr>
<tr>
<td>ipspace1</td>
<td>ipspace1</td>
<td>node1:e0d, node1:e0e, node2:e0d, node2:e0e</td>
</tr>
</tbody>
</table>

Creating a subnet

After you create the broadcast domain, you should create a subnet to allocate specific blocks of IPv4 or IPv6 addresses to be used later when you create LIFs for the SVM. This enables you to create LIFs more easily by specifying a subnet name instead of having to specify IP address and network mask values for each LIF.

Before you begin

You must be a cluster administrator to perform this task.

Steps

1. Create a subnet.

   **Example**

   ```bash
   network subnet create -broadcast-domain ipspace1 -ipspace ipspace1 -subnet-name ipspace1 -subnet 10.0.0.0/24 -gateway 10.0.0.1 -ip-ranges "10.0.0.128-10.0.0.130,10.0.0.132"
   ```

   The subnet name can be either a subnet IP value such as “192.0.2.0/24” or a string such as “ipspace1” like the one used in this example.

2. Verify that the subnet configuration is correct.
Example

The output from this example shows information about the subnet named “ipspace1” in the “ipspace1” IPspace. The subnet belongs to the broadcast domain name “ipspace1”. You can assign the IP addresses in this subnet to data LIFs for SVMs created in the “ipspace1” IPspace.

network subnet show –ipspace ipspace1

<table>
<thead>
<tr>
<th>IPspace:</th>
<th>Subnet</th>
<th>Broadcast Domain</th>
<th>Gateway</th>
<th>Avail/ Total</th>
<th>Ranges</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipspace1</td>
<td>10.0.0.0/24</td>
<td>ipspace1</td>
<td>10.0.0.1</td>
<td>4/4</td>
<td>10.0.0.128-10.0.0.130, 10.0.0.132</td>
</tr>
</tbody>
</table>

Creating an SVM

You must create an SVM to serve data to clients.

Before you begin

- You must be a cluster administrator to perform this task.
- You must know which security style the SVM root volume will have.

  If you plan to implement a Hyper-V or SQL Server over SMB solution on this SVM, you should use NTFS security style for the root volume. Volumes that contain Hyper-V files or SQL database files must be set to NTFS security at the time they are created. By setting the root volume security style to NTFS, you ensure that you do not inadvertently create UNIX or mixed security-style data volumes.

Steps

1. Determine which aggregates are candidates for containing the SVM root volume.

   storage aggregate show –has-mroot false

Example

storage aggregate show –has-mroot false

<table>
<thead>
<tr>
<th>Aggregate</th>
<th>Size</th>
<th>Available</th>
<th>Used%</th>
<th>State</th>
<th>#Vols</th>
<th>Nodes</th>
<th>RAID Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>aggr1</td>
<td>239.0GB</td>
<td>229.8GB</td>
<td>4%</td>
<td>online</td>
<td>4</td>
<td>node1</td>
<td>raid_dp, normal</td>
</tr>
<tr>
<td>aggr2</td>
<td>239.0GB</td>
<td>235.9GB</td>
<td>1%</td>
<td>online</td>
<td>2</td>
<td>node2</td>
<td>raid_dp, normal</td>
</tr>
<tr>
<td>aggr3</td>
<td>478.1GB</td>
<td>465.2GB</td>
<td>3%</td>
<td>online</td>
<td>1</td>
<td>node2</td>
<td>raid_dp, normal</td>
</tr>
</tbody>
</table>

The best practice is to use a separate aggregate for SVM root volumes. You should not create a root volume on an aggregate that contains data volumes.

You must choose an aggregate that has at least 1 GB of free space to contain the root volume. If you intend to configure NAS auditing on the SVM, you must have a minimum of 3 GB of extra free space on the root aggregate, with the extra space being used to create the auditing staging volume when auditing is enabled.

Note: If NAS auditing is already enabled on an existing SVM, the aggregate's staging volume is created immediately after aggregate creation is successfully completed.

2. Record the name of the aggregate on which you want to create the SVM root volume.
3. If you plan on specifying a language when you create the SVM and do not know the value to use, identify and record the value of the language you want to specify:

   `vserver create -language ?`

4. If you plan on specifying a Snapshot policy when you create the SVM and do not know the name of the policy, list the available policies and identify and record the name of the Snapshot policy you want to use:

   `volume snapshot policy show -vserver vserver_name`

5. If you plan on specifying a quota policy when you create the SVM and do not know the name of the policy, list the available policies and identify and record the name of the quota policy you want to use:

   `volume quota policy show -vserver vserver_name`

6. Create an SVM:

   ```
   vserver create -vserver vserver_name -aggregate aggregate_name -rootvolume root_volume_name -rootvolume-security-style {unix|ntfs|mixed} [-ipspace IPspace_name] [-language language] [-snapshot-policy snapshot_policy_name] [-quota-policy quota_policy_name] [-comment comment]
   ```

   **Example**

   ```
   vserver create -vserver vs1 -aggregate aggr3 -rootvolume vs1_root -rootvolume-security-style ntfs -ipspace ipspace1 -language en_US.UTF-8
   ```

   ![Vserver creation completed](image)

   7. Verify that the SVM configuration is correct.

   **Example**

   ```
   vserver show -vserver vs1
   ```

   ![Vserver configuration details](image)

   In this example, the command creates the SVM named “vs1” in IPspace “ipspace1”. The root volume is named “vs1_root” and is created on aggr3 with NTFS security style.
Related information
- Stretch MetroCluster installation and configuration
- Fabric-attached MetroCluster installation and configuration

Configuring LIFs on the SVM

An SVM serves data to clients through one or more network logical interfaces (LIFs). You must create LIFs on the ports you want to use to access data.

Before you begin
You must be a cluster administrator to perform this task.

About this task
You should not configure LIFs that carry CIFS traffic to automatically revert to their home nodes. This recommendation is mandatory if the CIFS server is to host a solution for nondisruptive operations with Hyper-V or SQL Server over SMB.

Steps
1. Determine which broadcast domain ports you want to use for the LIF.
   
   **Example**
   ```
   network port broadcast-domain show -ipspace ipspace1
   ```
   
<table>
<thead>
<tr>
<th>IPspace</th>
<th>Broadcast</th>
<th>Update</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipspace1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ipspace1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1500</td>
<td></td>
</tr>
<tr>
<td></td>
<td>node1:e0d</td>
<td>complete</td>
</tr>
<tr>
<td></td>
<td>node1:e0e</td>
<td>complete</td>
</tr>
<tr>
<td></td>
<td>node2:e0d</td>
<td>complete</td>
</tr>
<tr>
<td></td>
<td>node2:e0e</td>
<td>complete</td>
</tr>
</tbody>
</table>

2. Verify that the subnet you want to use for the LIFs contains sufficient unused IP addresses.
   
   **Example**
   ```
   network subnet show -ipspace ipspace1
   ```
   
<table>
<thead>
<tr>
<th>Subnet Name</th>
<th>Subnet</th>
<th>Broadcast</th>
<th>Gateway</th>
<th>Avail/Total</th>
<th>Ranges</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipspace1</td>
<td>10.0.0.0/24</td>
<td>ipspace1</td>
<td>10.0.0.1</td>
<td>4/4</td>
<td>10.0.0.128-10.0.0.130, 10.0.0.132</td>
</tr>
</tbody>
</table>

3. Create one or more LIFs on the ports you want to use to access data.
   
   **Example**
   ```
   network interface create -vserver vs1 -lif lif1 -role data -data-protocol nfs,cifs -home-node node1 -home-port e0d -subnet-name ipspace1
   network interface create -vserver vs1 -lif lif2 -role data -data-protocol nfs,cifs -home-node node2 -home-port e0d -subnet-name ipspace1
   ```

4. Verify that the LIF interface configuration is correct.
**Example**

```bash
network interface show -vserver vs1
```

<table>
<thead>
<tr>
<th>Vserver</th>
<th>Interface</th>
<th>Status</th>
<th>Network Address/Mask</th>
<th>Current Node</th>
<th>Current Port</th>
<th>Current Is Home</th>
</tr>
</thead>
<tbody>
<tr>
<td>vs1</td>
<td>lif1</td>
<td>up/up</td>
<td>10.0.0.128/24</td>
<td>node1</td>
<td>e0d</td>
<td>true</td>
</tr>
<tr>
<td></td>
<td>lif2</td>
<td>up/up</td>
<td>10.0.0.129/24</td>
<td>node2</td>
<td>e0d</td>
<td>true</td>
</tr>
</tbody>
</table>

5. Verify that the failover group configuration is as desired.

**Example**

```bash
network interface show -failover -vserver vs1
```

<table>
<thead>
<tr>
<th>Vserver</th>
<th>Interface</th>
<th>Home</th>
<th>Failover</th>
<th>Failover</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Policy</td>
<td>Group</td>
</tr>
<tr>
<td>vs1</td>
<td>lif1</td>
<td>node1:e0d</td>
<td>system-defined</td>
<td>ipspace1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Failover Targets: node1:e0d, node1:e0e, node2:e0d, node2:e0e</td>
<td></td>
</tr>
<tr>
<td></td>
<td>lif2</td>
<td>node2:e0d</td>
<td>system-defined</td>
<td>ipspace1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Failover Targets: node2:e0d, node2:e0e, node1:e0d, node1:e0e</td>
<td></td>
</tr>
</tbody>
</table>

**Configuring DNS services for the SVM**

You must configure DNS services for the SVM before creating an NFS or CIFS server. Generally, the DNS name servers are the Active Directory-integrated DNS servers for the domain that the NFS or CIFS server will join.

**About this task**

Active Directory-integrated DNS servers contain the service location records (SRV) for the domain LDAP and domain controller servers. If the SVM cannot find the Active Directory LDAP servers and domain controllers, NFS or CIFS server setup fails.

SVMs use the `hosts` name services `ns-switch` database to determine which name services to use and in which order when looking up information about hosts. The two supported name services for the `hosts` database are `files` and `dns`.

You must ensure that `dns` is one of the sources before you create the CIFS server.

**Note:** To view the statistics for DNS name services for the mgwd process and SecD process, use the Statistics UI.

**Steps**

1. Determine what the current configuration is for the `hosts` name services database.

**Example**

In this example, the `hosts` name service database uses the default settings.

```bash
vserver services name-service ns-switch show -vserver vs1 -database hosts
```
2. Perform the following actions, if required.
   a. Add the DNS name service to the `hosts` name service database in the desired order, or reorder the sources.

   **Example**
   In this example, the `hosts` database is configured to use DNS and local files in that order.
   ```
   vserver services name-service ns-switch modify -vserver vs1 -database hosts -sources dns, files
   ```

   b. Verify that the name services configuration is correct.

   **Example**
   ```
   vserver services name-service ns-switch show -vserver vs1 -database hosts
   ```

3. Configure DNS services.

   **Example**
   ```
   vserver services name-service dns create -vserver vs1 -domains example.com, example2.com -name-servers 10.0.0.50, 10.0.0.51
   ```

   **Note:** Starting in ONTAP 9.2, the `vserver services name-service dns create` command performs an automatic configuration validation and reports an error message if ONTAP is unable to contact the name server.

4. Verify that the DNS configuration is correct and that the service is enabled.

   **Example**
   ```
   Vserver: vs1
   Domains: example.com, example2.com
   Name Servers: 10.0.0.50, 10.0.0.51
   Enable/Disable DNS: enabled
   Timeout (secs): 2
   Maximum Attempts: 1
   ```

5. Validate the status of the name servers.

   **Example**
   ```
   cluster-1::> vserver services name-service dns check -vserver vs1
   ```

<table>
<thead>
<tr>
<th>Vserver</th>
<th>Name Server</th>
<th>Status</th>
<th>Status Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>vs1</td>
<td>10.0.0.50</td>
<td>up</td>
<td>Response time (msec): 2</td>
</tr>
<tr>
<td>vs1</td>
<td>10.0.0.51</td>
<td>up</td>
<td>Response time (msec): 2</td>
</tr>
</tbody>
</table>
The name service check command is available starting in ONTAP 9.2.

**Configuring dynamic DNS on the SVM**

If you want the Active Directory-integrated DNS server to dynamically register the DNS records of an NFS or CIFS server in DNS, you must configure dynamic DNS (DDNS) on the SVM.

**Before you begin**

DNS name services must be configured on the SVM. If you are using secure DDNS, you must use Active Directory-integrated DNS name servers and you must have created either an NFS or CIFS server or an Active Directory account for the SVM.

**About this task**

The specified FQDN must be unique.

**Note:** To avoid a configuration failure of an SVM FQDN that is not compliant to RFC rules for DDNS updates, use an FQDN name that is RFC compliant.

**Steps**

1. Configure DDNS on the SVM:

   ```
   vserver services name-service dns dynamic-update modify -vserver vserver_name -is-enabled true [-use-secure {true|false}] -vserver-fqdn FQDN_used_for_DNS_updates
   ```

   **Example**

   ```
   vserver services name-service dns dynamic-update modify -vserver vs1 -is-enabled true -use-secure true -vserver-fqdn vs1.example.com
   ```

   **Note:** Asterisks cannot be used as part of the customized FQDN. For example, *.*.netapp.com is not valid.

2. Verify that the DDNS configuration is correct:

   ```
   vserver services name-service dns dynamic-update show
   ```

   **Example**

   ```
   vserver services name-service dns dynamic-update show
   ```

<table>
<thead>
<tr>
<th>Vserver</th>
<th>Is-Enabled</th>
<th>Use-Secure</th>
<th>Vserver FQDN</th>
<th>TTL</th>
</tr>
</thead>
<tbody>
<tr>
<td>vs1</td>
<td>true</td>
<td>true</td>
<td>vs1.example.com</td>
<td>24h</td>
</tr>
</tbody>
</table>
Configuring network ports (cluster administrators only)

Ports are either physical ports (NICs) or virtualized ports, such as interface groups or VLANs.

Types of network ports

The network ports are either physical ports or virtualized ports. Virtual local area networks (VLANs) and interface groups constitute the virtual ports. Interface groups treat several physical ports as a single port, while VLANs subdivide a physical port into multiple separate logical ports.

physical ports

LIFs can be configured directly on physical ports.

interface group

A port aggregate containing two or more physical ports that act as a single trunk port. An interface group can be single-mode, multimode, or dynamic multimode.

VLAN

A logical port that receives and sends VLAN-tagged (IEEE 802.1Q standard) traffic. VLAN port characteristics include the VLAN ID for the port. The underlying physical port or interface group ports are considered VLAN trunk ports, and the connected switch ports must be configured to trunk the VLAN IDs.

The underlying physical port or interface group ports for a VLAN port can continue to host LIFs, which transmit and receive untagged traffic.

virtual IP (VIP) port

A logical port that is used as the home port for a VIP LIF. VIP ports are created automatically by the system and support only a limited number of operations. VIP ports are supported starting with ONTAP 9.5.

The port naming convention is `enumberletter`:

- The first character describes the port type.
  - “e” represents Ethernet.
- The second character indicates the numbered slot in which the port adapter is located.
- The third character indicates the port's position on a multiport adapter.
  - “a” indicates the first port, “b” indicates the second port, and so on.

For example, `e0b` indicates that an Ethernet port is the second port on the node's motherboard.

VLANs must be named by using the syntax `port_name-vlan-id`.

- `port_name` specifies the physical port or interface group and
- `vlan-id` specifies the VLAN identification on the network. For example, `e1c-80` is a valid VLAN name.

Combining physical ports to create interface groups

An interface group is created by combining two or more physical ports into a single logical port. The logical port provides increased resiliency, increased availability, and load sharing.
Interface group types

Three types of interface groups are supported on the storage system: single-mode, static multimode, and dynamic multimode. Each interface group provides different levels of fault tolerance. Multimode interface groups provide methods for load balancing network traffic.

Characteristics of single-mode interface groups

In a single-mode interface group, only one of the interfaces in the interface group is active. The other interfaces are on standby, ready to take over if the active interface fails.

Characteristics of a single-mode interface groups:

- For failover, the cluster monitors the active link and controls failover. Because the cluster monitors the active link, there is no switch configuration required.
- There can be more than one interface on standby in a single-mode interface group.
- If a single-mode interface group spans multiple switches, you must connect the switches with an Inter-Switch link (ISL).
- For a single-mode interface group, the switch ports must be in the same broadcast domain.
- Link-monitoring ARP packets, which have a source address of 0.0.0.0, are sent over the ports to verify that the ports are in the same broadcast domain.

The following figure is an example of a single-mode interface group. In the figure, e0a and e1a are part of the a0a single-mode interface group. If the active interface, e0a, fails, the standby e1a interface takes over and maintains the connection to the switch.

![Diagram of single-mode interface group]

Note: To accomplish single-mode functionality, the recommended approach is to instead use failover groups. By using a failover group, the second port can still be used for other LIFs and need not remain unused. Additionally, failover groups can span more than two ports and can span ports on multiple nodes.

Characteristics of static multimode interface groups

The static multimode interface group implementation in ONTAP complies with IEEE 802.3ad (static). Any switch that supports aggregates, but does not have control packet exchange for configuring an aggregate, can be used with static multimode interface groups.

Static multimode interface groups do not comply with IEEE 802.3ad (dynamic), also known as Link Aggregation Control Protocol (LACP). LACP is equivalent to Port Aggregation Protocol (PAgP), the proprietary link aggregation protocol from Cisco.

The following are characteristics of a static multimode interface group:

- All interfaces in the interface group are active and share a single MAC address.
Multiple individual connections are distributed among the interfaces in the interface group.

Each connection or session uses one interface within the interface group.

When you use the sequential load balancing scheme, all sessions are distributed across available links on a packet-by-packet basis, and are not bound to a particular interface from the interface group.

- Static multimode interface groups can recover from a failure of up to “n-1” interfaces, where n is the total number of interfaces that form the interface group.
- If a port fails or is unplugged, the traffic that was traversing the failed link is automatically redistributed to one of the remaining interfaces.
- Static multimode interface groups can detect a loss of link, but they cannot detect a loss of connectivity to the client or switch misconfigurations that might impact connectivity and performance.
- A static multimode interface group requires a switch that supports link aggregation over multiple switch ports.
  The switch is configured so that all ports to which links of an interface group are connected are part of a single logical port. Some switches might not support link aggregation of ports configured for jumbo frames. For more information, see your switch vendor's documentation.
- Several load balancing options are available to distribute traffic among the interfaces of a static multimode interface group.

The following figure is an example of a static multimode interface group. Interfaces e0a, e1a, e2a, and e3a are part of the a1a multimode interface group. All four interfaces in the a1a multimode interface group are active.

Several technologies exist that enable traffic in a single aggregated link to be distributed across multiple physical switches. The technologies used to enable this capability vary among networking products. Static multimode interface groups in ONTAP conform to the IEEE 802.3 standards. If a particular multiple switch link aggregation technology is said to interoperate with or conform to the IEEE 802.3 standards, it should operate with ONTAP.

The IEEE 802.3 standard states that the transmitting device in an aggregated link determines the physical interface for transmission. Therefore, ONTAP is only responsible for distributing outbound traffic, and cannot control how inbound frames arrive. If you want to manage or control the transmission of inbound traffic on an aggregated link, that transmission must be modified on the directly connected network device.
Dynamic multimode interface group

Dynamic multimode interface group implement Link Aggregation Control Protocol (LACP) to communicate group membership to the directly attached switch. LACP enables you to detect the loss of link status and the inability of the node to communicate with the direct-attached switch port.

Dynamic multimode interface group implementation in ONTAP complies with IEEE 802.3 AD (802.1 AX). ONTAP does not support Port Aggregation Protocol (PAgP), which is a proprietary link aggregation protocol from Cisco.

A dynamic multimode interface group requires a switch that supports LACP.

ONTAP implements LACP in nonconfigurable active mode that works well with switches that are configured in either active or passive mode. ONTAP implements the long and short LACP timers (for use with nonconfigurable values 3 seconds and 90 seconds), as specified in IEEE 802.3 AD (802.1AX).

The ONTAP load balancing algorithm determines the member port to be used to transmit outbound traffic, and does not control how inbound frames are received. The switch determines the member (individual physical port) of its port channel group to be used for transmission, based on the load balancing algorithm configured in the switch’s port channel group. Therefore, the switch configuration determines the member port (individual physical port) of the storage system to receive traffic. For more information about configuring the switch, see the documentation from your switch vendor.

If an individual interface fails to receive successive LACP protocol packets, then that individual interface is marked as lag_inactive in the output of ifgrp status command. Existing traffic is automatically rerouted to any remaining active interfaces.

The following rules apply when using dynamic multimode interface groups:

- Dynamic multimode interface groups should be configured to use the port-based, IP-based, MAC-based, or round robin load balancing methods.

- In a dynamic multimode interface group, all interfaces must be active and share a single MAC address.

The following figure is an example of a dynamic multimode interface group. Interfaces e0a, e1a, e2a, and e3a are part of the a1a multimode interface group. All four interfaces in the a1a dynamic multimode interface group are active.

![Dynamic multimode interface group example](image)

Load balancing in multimode interface groups

You can ensure that all interfaces of a multimode interface group are equally utilized for outgoing traffic by using the IP address, MAC address, sequential, or port-based load balancing methods to distribute network traffic equally over the network ports of a multimode interface group.

The load balancing method for a multimode interface group can be specified only when the interface group is created.
**IP address and MAC address load balancing**

IP address and MAC address load balancing are the methods for equalizing traffic on multimode interface groups.

These load balancing methods use a fast hashing algorithm on the source and destination addresses (IP address and MAC address). If the result of the hashing algorithm maps to an interface that is not in the UP link-state, the next active interface is used.

**Note:** Do not select the MAC address load balancing method when creating interface groups on a system that connects directly to a router. In such a setup, for every outgoing IP frame, the destination MAC address is the MAC address of the router. As a result, only one interface of the interface group is used.

IP address load balancing works in the same way for both IPv4 and IPv6 addresses.

**Sequential load balancing**

You can use sequential load balancing to equally distribute packets among multiple links using a round robin algorithm. You should use the sequential option for load balancing a single connection's traffic across multiple links to increase single connection throughput. However, this method might cause out-of-order packet delivery.

If the remote TCP endpoints do not handle TCP reassembly correctly or lack enough memory to store out-of-order packets, they might be forced to drop packets. Therefore, this might result in unnecessary retransmissions from the storage controller.

**Port-based load balancing**

You can equalize traffic on a multimode interface group based on the transport layer (TCP/UDP) ports by using the port-based load balancing method.

The port-based load balancing method uses a fast hashing algorithm on the source and destination IP addresses along with the transport layer port number.

**Creating an interface group**

You can create an interface group—single-mode, static multimode, or dynamic multimode (LACP)—to present a single interface to clients by combining the capabilities of the aggregated network ports.

**About this task**

- In a single-mode interface group, you can select the active port or designate a port as nonfavored by running the `ifgrp` command from the nodeshell.

- When creating a multimode interface group, you can specify any of the following load-balancing methods:
  - `mac`: Network traffic is distributed on the basis of MAC addresses.
  - `ip`: Network traffic is distributed on the basis of IP addresses.
  - `sequential`: Network traffic is distributed as it is received.
  - `port`: Network traffic is distributed on the basis of the transport layer (TCP/UDP) ports.

**Note:** The MAC address of an `ifgrp` is determined by the order of the underlying ports and how these ports initialize during bootup. You should therefore not assume that the `ifgrp` MAC address is persistent across reboots or ONTAP upgrades.
Step

1. Use the `network port ifgrp create` command to create an interface group.
   Interface groups must be named using the syntax `a<number><letter>`. For example, a0a, a0b, a1c, and a2a are valid interface group names.
   For more information about this command, see the man pages.

   **Example**
   The following example shows how to create an interface group named a0a with a distribution function of ip and a mode of multimode:
   ```
   cluster-1::> network port ifgrp create -node cluster-1-01 -ifgrp a0a -distr-func ip -mode multimode
   ```

### Adding a port to an interface group

You can add up to 16 physical ports to an interface group for all port speeds.

**Before you begin**

**Step**

1. Add network ports to the interface group:
   ```
   network port ifgrp add-port
   ```
   For more information about this command, see the man page.

   **Example**
   The following example shows how to add port e0c to an interface group named a0a:
   ```
   cluster-1::> network port ifgrp add-port -node cluster-1-01 -ifgrp a0a -port e0c
   ```

### Removing a port from an interface group

You can remove a port from an interface group that hosts LIFs, as long as it is not the last port in the interface group. There is no requirement that the interface group must not host LIFs or that the interface group must not be the home port of a LIF considering that you are not removing the last port from the interface group. However, if you are removing the last port, then you must migrate or move the LIFs from the interface group first.

**About this task**

You can remove up to 16 ports (physical interfaces) from an interface group.

**Step**

1. Remove network ports from an interface group:
   ```
   network port ifgrp remove-port
   ```

   **Example**
   The following example shows how to remove port e0c from an interface group named a0a:
Deleting an interface group

You can delete interface groups if you want to configure LIFs directly on the underlying physical ports or decide to change the interface group mode or distribution function.

Before you begin

- The interface group must not be hosting a LIF.
- The interface group must be neither the home port nor the failover target of a LIF.

Step

1. Use the `network port ifgrp delete` command to delete an interface group.

   For more information about this command, see the man pages.

Example

The following example shows how to delete an interface group named a0b:

```
cluster-1::> network port ifgrp delete -node cluster-1-01 -ifgrp a0b
```

Related tasks

- [Modifying network port attributes](#)
- [Displaying LIF information](#)

Configuring VLANs over physical ports

VLANs provide logical segmentation of networks by creating separate broadcast domains that are defined on a switch port basis as opposed to the traditional broadcast domains, defined on physical boundaries. A VLAN can span multiple physical network segments. The end-stations belonging to a VLAN are related by function or application.

For example, end-stations in a VLAN might be grouped by departments, such as engineering and accounting, or by projects, such as release1 and release2. Because physical proximity of the end-stations is not essential in a VLAN, you can disperse the end-stations geographically and still contain the broadcast domain in a switched network.

You can manage VLANs by creating, deleting, or displaying information about them.

Note: You should not create a VLAN on a network interface with the same identifier as the native VLAN of the switch. For example, if the network interface e0b is on native VLAN 10, you should not create a VLAN e0b-10 on that interface.

Related tasks

- [Creating a VLAN](#)
Creating a VLAN

You can create a VLAN for maintaining separate broadcast domains within the same network domain by using the `network port vlan create` command.

**Before you begin**

Your network administrator must have confirmed that the following requirements have been met:

- The switches deployed in the network must either comply with IEEE 802.1Q standards or have a vendor-specific implementation of VLANs.
- For supporting multiple VLANs, an end-station must be statically configured to belong to one or more VLANs.
- The VLAN is not attached to a port hosting a cluster LIF.
- The VLAN is not attached to ports assigned to the Cluster IPspace.
- The VLAN is not created on an interface group port that contains no member ports.

**About this task**

In certain circumstances, if you want to create the VLAN port on a degraded port without correcting the hardware issue or any software misconfiguration, then you can set the `-ignore-health-status` parameter of the `network port modify` command as `true`.

Creating a VLAN attaches the VLAN to the network port on a specified node in a cluster.

When you configure a VLAN over a port for the first time, the port might go down, resulting in a temporary disconnection of the network. Subsequent VLAN additions to the same port do not affect the port state.

**Note:** You should not create a VLAN on a network interface with the same identifier as the native VLAN of the switch. For example, if the network interface e0b is on native VLAN 10, you should not create a VLAN e0b-10 on that interface.

**Step**

1. Use the `network port vlan create` command to create a VLAN.

   You must specify either the `vlan-name` or the `port` and `vlan-id` options when creating a VLAN. The VLAN name is a combination of the name of the port (or interface group) and the network switch VLAN identifier, with a hyphen in between. For example, e0c-24 and e1c-80 are valid VLAN names.

**Example**

The following example shows how to create a VLAN e1c-80 attached to network port e1c on the node cluster-1-01:

```
cluster-1::> network port vlan create -node cluster-1-01 -vlan-name e1c-80
```

For more information about this command, see the man page.
Deleting a VLAN

You might have to delete a VLAN before removing a NIC from its slot. When you delete a VLAN, it is automatically removed from all of the failover rules and groups that use it.

Before you begin

There must be no LIFs associated with the VLAN.

About this task

Deletion of the last VLAN from a port might cause a temporary disconnection of the network from the port.

Step

1. Use the `network port vlan delete` command to delete a VLAN.

   Example

   The following example shows how to delete VLAN e1c-80 from network port e1c on the node cluster-1-01:

   ```
   cluster-1::> network port vlan delete -node cluster-1-01 -vlan-name e1c-80
   ```

Related tasks

Displaying LIF information on page 121

Modifying network port attributes

You can modify the autonegotiation, duplex, flow control, speed, and health settings of a physical network port.

Before you begin

The port that you want to modify must not be hosting any LIFs.

About this task

- You should not modify the administrative settings of the 100 GbE, 40 GbE, 10 GbE or 1 GbE network interfaces. The values that you can set for duplex mode and port speed are referred to as administrative settings. Depending on network limitations, the administrative settings can differ from the operational settings (that is, the duplex mode and speed that the port actually uses).

- You should not modify the administrative settings of the underlying physical ports in an interface group. The `-up-admin` parameter (available at the advanced privilege level) modifies the administrative settings of the port.

- The `-up-admin` administrative setting should not be set to `false` for all ports on a node, or for the port that hosts the last operational cluster LIF on a node.

- You should not modify the MTU size of the management port, e0M.
• The MTU size of a port in a broadcast domain cannot be changed from the MTU value that is set for the broadcast domain.

• The MTU size of a VLAN cannot exceed the value of the MTU size of its base port.

Step

1. Modify the attributes of a network port:

   ```
   network port modify
   ```

   You can set the `-ignore-health-status` field to `true` for specifying that the system can ignore the network port health status of a specified port. The network port health status is automatically changed from degraded to healthy, and this port can now be used for hosting LIFs. You should set the flow control of cluster ports to `none`. By default, the flow control is set to `full`.

   **Example**

   The following command disables the flow control on port e0b by setting the flow control to `none`:

   ```
   cluster-1::> network port modify -node cluster-1-01 -port e0b -flowcontrol-admin none
   ```

Modifying MTU setting for interface group ports

To modify the MTU setting for interface groups, you must modify the MTU of the broadcast domain.

Steps

1. Modify the broadcast domain settings:

   ```
   broadcast-domain modify -broadcast-domain broadcast_domain_name -mtu mtu_setting
   ```

   The following warning message is displayed:

   ```
   Warning: Changing broadcast domain settings will cause a momentary data-serving interruption.
   Do you want to continue? {y|n}: y
   ```

2. Enter `y` to continue.

3. Verify that the MTU setting were modified correctly:

   ```
   net port show
   ```

   **Example**

   ```
   vsim::*> net port show
   (network port show)
   ```

   Node: vsim-01

   Ignore

   Health Health Speed(Mbps)
   Port IPspace Broadcast Domain Link MTU Admin/Oper
   Status Status
   --------- ------- --------------- ---- ------ -----------
Monitoring the health of network ports

ONTAP management of network ports includes automatic health monitoring and a set of health monitors to help you identify network ports that might not be suitable for hosting LIFs.

About this task

If a health monitor determines that a network port is unhealthy, it warns administrators through an EMS message or marks the port as degraded. ONTAP avoids hosting LIFs on degraded network ports if there are healthy alternative failover targets for that LIF. A port can become degraded because of a soft failure event, such as link flapping (links bouncing quickly between up and down) or network partitioning:

- Network ports in the cluster IPspace are marked as degraded when they experience link flapping or loss of Layer-2 reachability to other network ports in the broadcast domain.

- Network ports in non-cluster IPspaces are marked as degraded when they experience link flapping.

You must be aware of the following behaviors of a degraded port:

- A degraded port cannot be included in a VLAN or an interface group. If a member port of an interface group is marked as degraded, but the interface group is still marked as healthy, LIFs can be hosted on that interface group.

- LIFs are automatically migrated from degraded ports to healthy ports. During a failover event, a degraded port is not considered as the failover target. If no healthy ports are available, degraded ports host LIFs according to the normal failover policy.

- You cannot create, migrate, or revert a LIF to a degraded port. You can modify the ignore-health-status setting of the network port to true. You can then host a LIF on the healthy ports.

Steps

1. Log in to the advanced privilege mode:
   ```bash
   set -privilege advanced
   ```

2. Check which health monitors are enabled for monitoring network port health:
   ```bash
   network options port-health-monitor show
   ```

   The health status of a port is determined by the value of health monitors. The following health monitors are available and enabled by default in ONTAP:

   - Link-flapping health monitor: Monitors link flapping
     If a port has link flapping more than once in five minutes, this port is marked as degraded.

   - L2 reachability health monitor: Monitors whether all ports configured in the same broadcast domain have L2 reachability to each other
This health monitor reports L2 reachability issues in all IPspaces; however, it marks only the ports in the cluster IPspace as degraded.

- **CRC monitor**: Monitors the CRC statistics on the ports
  This health monitor does not mark a port as degraded, but generates an EMS message when a very high CRC failure rate is observed.

**Example**

```
cluster-1::*> network options port-health-monitor show
IPspace         Enabled Port Health Monitors
-------------   ----------------------------
Cluster         l2_reachability, link_flapping
Default         l2_reachability, link_flapping
2 entries were displayed.
```

3. Enable or disable any of the health monitors for an IPspace as desired by using the `network options port-health-monitor modify` command.

4. View the detailed health of a port:
   ```
   network port show -health
   ```
   The command output displays the health status of the port, ignore health status setting, and list of reasons the port is marked as degraded. A port health status can be healthy or degraded. If the ignore health status setting is `true`, it indicates that the port health status has been modified from degraded to healthy by the administrator. If the ignore health status setting is `false`, the port health status is determined automatically by the system.

**Example**

```
cluster-1::> network port show -health
Ignore Health   Health
Node       Port      Link Status   Status Degraded Reasons
--------   --------- ---- -------- ------ -----------------------
node1
   e0a       up   healthy  false  -
   e0b       up   healthy  false  -
   e0c       up   degraded false  l2_reachability, link_flapping
   e0d       up   degraded false  l2_reachability

node2
   e0a       up   healthy  false  -
   e0b       up   healthy  false  -
   e0c       up   healthy  false  -
   e0d       up   degraded false  -
8 entries were displayed.
```

**Converting 40GbE NIC ports into multiple 10GbE ports for 10GbE connectivity**

The X1144A-R6 and the X91440A-R6 40GbE Network Interface Cards (NICs) can be converted to support four 10GbE ports. If you are connecting a hardware platform that supports one of these NICs
to a cluster that supports 10GbE cluster interconnect and customer data connections, the NIC must be converted to provide the necessary 10GbE connections.

**Before you begin**
You must be using a supported breakout cable.

**About this task**
The following hardware platforms support the X1144A-R6 NIC.

- FAS8200
- AFF A300
- AFF A700s

**Note:** On the X1144A-R6 NIC, only port A can be converted to support the four 10GbE connections. Once port A is converted, port e is not available for use.

The following hardware platforms support the X91440A-R6 NIC.

- FAS9000
- AFF A700

**Steps**
1. Enter maintenance mode.
2. Convert the NIC from 40GbE support to 10GbE support.
   ```
   nicadmin convert -m [40G | 10G] [port-name]
   ```
3. Reboot the system.

**Removing a NIC from the node**
You might have to remove a faulty NIC from its slot or move the NIC to another slot for maintenance purposes.

**Before you begin**
- All LIFs hosted on the NIC ports must have been migrated or deleted.
- None of the NIC ports can be the home ports of any LIFs.
- You must have advanced privileges to delete the ports from a NIC.

**Steps**
1. Delete the ports from the NIC:
   ```
   network port delete
   ```
2. Verify that the ports have been deleted:
   ```
   network port show
   ```
3. Repeat step 1, if the output of the `network port show` command still shows the deleted port.
Related information

- Replacing an I/O module
- Moving or replacing a NIC in clustered Data ONTAP 8.1 or later
Configuring IPspaces (cluster administrators only)

IPspaces enable you to configure a single ONTAP cluster so that it can be accessed by clients from more than one administratively separate network domain, even if those clients are using the same IP address subnet range. This allows for separation of client traffic for privacy and security.

An IPspace defines a distinct IP address space in which storage virtual machines (SVMs) reside. Ports and IP addresses defined for an IPspace are applicable only within that IPspace. A distinct routing table is maintained for each SVM within an IPspace; therefore, no cross-SVM or cross-IPspace traffic routing occurs.

**Note:** IPspaces support both IPv4 and IPv6 addresses on their routing domains.

If you are managing storage for a single organization, then you do not need to configure IPspaces. If you are managing storage for multiple companies on a single ONTAP cluster, and you are certain that none of your customers have conflicting networking configurations, then you also do not need to use IPspaces. In many cases, the use of storage virtual machines (SVMs), with their own distinct IP routing tables, can be used to segregate unique networking configurations instead of using IPspaces.

**Related tasks**

*Creating IPspaces* on page 45

Example of using IPspaces

A common application for using IPspaces is when a Storage Service Provider (SSP) needs to connect customers of companies A and B to an ONTAP cluster on the SSP’s premises and both companies are using the same private IP address ranges.

The SSP creates SVMs on the cluster for each customer and provides a dedicated network path from two SVMs to company A’s network and from the other two SVMs to company B’s network.

This type of deployment is shown in the following illustration, and it works if both companies use non-private IP address ranges. However, the illustration shows both companies using the same private IP address ranges, which causes problems.
Both companies use the private IP address subnet 10.0.0.0, causing the following problems:

- The SVMs in the cluster at the SSP location have conflicting IP addresses if both companies decide to use the same IP address for their respective SVMs.

- Even if the two companies agree on using different IP addresses for their SVMs, problems can arise. For example, if any client in A’s network has the same IP address as a client in B’s network, packets destined for a client in A’s address space might get routed to a client in B’s address space, and vice versa.

- If the two companies decide to use mutually exclusive address spaces (for example, A uses 10.0.0.0 with a network mask of 255.128.0.0 and B uses 10.128.0.0 with a network mask of 255.128.0.0), the SSP needs to configure static routes on the cluster to route traffic appropriately to A’s and B’s networks.
  This solution is neither scalable (because of static routes) nor secure (broadcast traffic is sent to all interfaces of the cluster).

To overcome these problems, the SSP defines two IPspaces on the cluster—one for each company. Because no cross-IPspace traffic is routed, the data for each company is securely routed to its respective network even if all of the SVMs are configured in the 10.0.0.0 address space, as shown in the following illustration:
Additionally, the IP addresses referred to by the various configuration files, such as the /etc/hosts file, the /etc/hosts.equiv file, and the /etc/rc file, are relative to that IPspace. Therefore, the IPspaces enable the SSP to configure the same IP address for the configuration and authentication data for multiple SVMs, without conflict.

**Standard properties of IPspaces**

Special IPspaces are created by default when the cluster is first created. Additionally, special storage virtual machines (SVMs) are created for each IPspace.

Two IPspaces are created automatically when the cluster is initialized:

- **“Default” IPspace**
  This IPspace is a container for ports, subnets, and SVMs that serve data. If your configuration does not need separate IPspaces for clients, all SVMs can be created in this IPspace. This IPspace also contains the cluster management and node management ports.

- **“Cluster” IPspace**
  This IPspace contains all cluster ports from all nodes in the cluster. It is created automatically when the cluster is created. It provides connectivity to the internal private cluster network. As additional nodes join the cluster, cluster ports from those nodes are added to the “Cluster” IPspace.
A “system” SVM exists for each IPspace. When you create an IPspace, a default system SVM of the same name is created:

- The system SVM for the “Cluster” IPspace carries cluster traffic between nodes of a cluster on the internal private cluster network.
  It is managed by the cluster administrator, and it has the name “Cluster”.

- The system SVM for the “Default” IPspace carries management traffic for the cluster and nodes, including the intercluster traffic between clusters.
  It is managed by the cluster administrator, and it uses the same name as the cluster.

- The system SVM for a custom IPspace that you create carries management traffic for that SVM.
  It is managed by the cluster administrator, and it uses the same name as the IPspace

One or more SVMs for clients can exist in an IPspace. Each client SVM has its own data volumes and configurations, and it is administered independently of other SVMs.

Creating IPspaces

IPspaces are distinct IP address spaces in which storage virtual machines (SVMs) reside. You can create IPspaces when you need your SVMs to have their own secure storage, administration, and routing.

About this task

Beginning with ONTAP 9, there is a cluster-wide limit of 512 IPspaces. The cluster-wide limit is reduced to 256 IPspaces for clusters that contain nodes with 6 GB of RAM or less for platforms such as FAS2220 or FAS2240. See the Hardware Universe to determine whether additional limits apply to your platform.

NetApp Hardware Universe

Note: An IPspace name cannot be “all” because “all” is a system-reserved name.

Step

1. Create an IPspace:

   ```
   network ipspace create -ipspace ipspace_name
   ```

   `ipspace_name` is the name of the IPspace that you want to create.

   Example

   The following command creates the IPspace `ipspace1` on a cluster:

   ```
   cluster-1:~> network ipspace create -ipspace ipspace1
   ```

After you finish

If you create an IPspace in a cluster with a MetroCluster configuration, IPspace objects must be manually replicated to the partner clusters. Any SVMs that are created and assigned to an IPspace before the IPspace is replicated will not be replicated to the partner clusters.

After creating an IPspace, you must create a broadcast domain to define the ports that will be part of the IPspace prior to creating SVMs.

Related tasks

   Creating a broadcast domain on page 49
Displaying IPspaces

You can display the list of IPspaces that exist in a cluster, and you can view the storage virtual machines (SVMs), broadcast domains, and ports that are assigned to each IPspace.

Step

1. Display the IPspaces and SVMs in a cluster:

   `network ipspace show [-ipspace ipspace_name]`

Example

The following command displays all of the IPspaces, SVMs, and broadcast domains in the cluster:

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

<table>
<thead>
<tr>
<th>IPspace</th>
<th>Vserver List</th>
<th>Broadcast Domains</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cluster</td>
<td>Cluster</td>
<td>Cluster</td>
</tr>
<tr>
<td>Default</td>
<td>vs1, cluster-1</td>
<td>Default</td>
</tr>
<tr>
<td>ipspace1</td>
<td>vs3, vs4, ipspace1</td>
<td>bcast1</td>
</tr>
</tbody>
</table>
```

The following command displays the nodes and ports that are part of IPspace ipspace1:

```
cluster-1::> network ipspace show -ipspace ipspace1

IPspace name: ipspace1
  Ports: cluster-1-01:e0c, cluster-1-01:e0d, cluster-1-02:e0c, cluster-1-02:e0d, cluster-1-02:e0e
  Broadcast Domains: bcast1
  Vservers: vs3, vs4, ipspace1
```

Deleting an IPspace

If you no longer need an IPspace, you can delete it.

Before you begin

There must be no broadcast domains, network interfaces, or SVMs associated with the IPspace you want to delete.

The system-defined “Default” and “Cluster” IPspaces cannot be deleted.

Step

1. Delete an IPspace:

   `network ipspace delete -ipspace ipspace_name`

Example

The following command deletes IPspace ipspace1 from the cluster:

```
```
cluster-1::> network ipspace delete -ipspace ipspacel
Configuring broadcast domains (cluster administrators only)

Broadcast domains enable you to group network ports that belong to the same layer 2 network. The ports in the group can then be used by a storage virtual machine (SVM) for data or management traffic.

A broadcast domain resides in an IPspace. During cluster initialization, the system creates two default broadcast domains:

- The “Default” broadcast domain contains ports that are in the “Default” IPspace. These ports are used primarily to serve data. Cluster management and node management ports are also in this broadcast domain.
- The “Cluster” broadcast domain contains ports that are in the “Cluster” IPspace. These ports are used for cluster communication and include all cluster ports from all nodes in the cluster.

If you have created unique IPspaces to separate client traffic, then you need to create a broadcast domain in each of those IPspaces.

Related tasks

Creating a broadcast domain on page 49

Example of using broadcast domains

A common application for using broadcast domains is when a system administrator wants to reserve specific ports for use by a certain client or group of clients. A broadcast domain should include ports from many nodes in the cluster to provide high availability for the connections to SVMs.

The illustration shows the ports assigned to three broadcast domains in a four-node cluster:

- The “Cluster” broadcast domain is created automatically during cluster initialization, and it contains ports a and b from each node in the cluster.
- The “Default” broadcast domain is also created automatically during cluster initialization, and it contains ports c and d from each node in the cluster.
- The bcast1 broadcast domain has been created manually, and it contains ports e, f, and g from each node in the cluster.

This broadcast domain was created by the system administrator specifically for a new client to access data through a new SVM.
A failover group of the same name and with the same network ports as each of the broadcast domains is created automatically. This failover group is automatically managed by the system, meaning that as ports are added or removed from the broadcast domain, they are automatically added or removed from this failover group.

Creating a broadcast domain

You create a broadcast domain to group network ports in the cluster that belong to the same layer 2 network. The ports can then be used by SVMs.

Before you begin

The ports you plan to add to the broadcast domain must not belong to another broadcast domain.

About this task

• All broadcast domain names must be unique within an IPspace.
• The ports added to a broadcast domain can be physical network ports, VLANs, or interface groups (ifgrps).
• If the ports you want to use belong to another broadcast domain, but are unused, you use the network port broadcast-domain remove-ports command to remove the ports from the existing broadcast domain.
• The MTU of the ports added to a broadcast domain are updated to the MTU value set in the broadcast domain.
• The MTU value must match all of the devices connected to that layer 2 network.
• If you do not specify an IPspace name, the broadcast domain is created in the “Default” IPspace.

To make system configuration easier, a failover group of the same name is created automatically that contains the same ports.

Steps

1. View the ports that are not currently assigned to a broadcast domain:
   network port show
   If the display is large, use the network port show -broadcast-domain command to view only unassigned ports.

2. Create a broadcast domain:
   network port broadcast-domain create -broadcast-domain
   broadcast_domain_name -mtu mtu_value [-ipspace ipspace_name] [-ports ports_list]

   • broadcast_domain_name is the name of the broadcast domain you want to create.
   • mtu_value is the MTU size for IP packets; 1500 and 9000 are typical values.
   • This value is applied to all ports that are added to this broadcast domain.
   • ipspace_name is the name of the IPspace to which this broadcast domain will be added.
     The “Default” IPspace is used unless you specify a value for this parameter.
   • ports_list is the list of ports that will be added to the broadcast domain.
     The ports are added in the format node_name:port_number, for example, node1:e0c.
3. Verify that the broadcast domain was created as desired:

```
network port show -instance -broadcast-domain new_domain
```

**Example**

The following command creates broadcast domain bcast1 in the Default IPspace, sets the MTU to 1500, and adds four ports:

```
cluster-1::> network port broadcast-domain create -broadcast-domain bcast1 -mtu 1500 -ports
cluster1-01:e0e,cluster1-01:e0f,cluster1-02:e0e, cluster1-02:e0f
```

**After you finish**

You can define the pool of IP addresses that will be available in the broadcast domain by creating a subnet, or you can assign SVMs and interfaces to the IPspace at this time. For more information, see the *Cluster Peering Express Guide*.

If you need to change the name of an existing broadcast domain, you use the `network port broadcast-domain rename` command.

**Related tasks**

- *Adding or removing ports from a broadcast domain* on page 50
- *Merging broadcast domains* on page 52
- *Splitting broadcast domains* on page 52
- *Creating IPspaces* on page 45
- *Creating a subnet* on page 60
- *Adding or removing ports from a broadcast domain* on page 50
- *Changing the MTU value for ports in a broadcast domain* on page 53

**Related information**

*Cluster and SVM peering express configuration*

---

**Adding or removing ports from a broadcast domain**

You can add network ports when initially creating a broadcast domain, or you can add ports to, or remove ports from, a broadcast domain that already exists. This allows you to efficiently use all the ports in the cluster.

**Before you begin**

- Ports you plan to add to a broadcast domain must not belong to another broadcast domain.
- Ports that already belong to an interface group cannot be added individually to a broadcast domain.

**About this task**

The following rules apply when adding and removing network ports:

<table>
<thead>
<tr>
<th>When adding ports...</th>
<th>When removing ports...</th>
</tr>
</thead>
<tbody>
<tr>
<td>The ports can be network ports, VLANs, or interface groups (ifgrps).</td>
<td>N/A</td>
</tr>
</tbody>
</table>
When adding ports...

<table>
<thead>
<tr>
<th>When removing ports...</th>
</tr>
</thead>
<tbody>
<tr>
<td>The ports are added to the system-defined failover group of the broadcast domain.</td>
</tr>
<tr>
<td>The MTU of the ports is updated to the MTU value set in the broadcast domain.</td>
</tr>
<tr>
<td>The IPspace of the ports is updated to the IPspace value of the broadcast domain.</td>
</tr>
</tbody>
</table>

Note: If you remove the last member port of an interface group using the `network port ifgrp remove-port` command, it causes the interface group port to be removed from the broadcast domain because an empty interface group port is not allowed in a broadcast domain.

Steps

1. Display the ports that are currently assigned or unassigned to a broadcast domain by using the `network port show` command.

2. Add or remove network ports from the broadcast domain:

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>Use...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add ports to a broadcast domain</td>
<td><code>network port broadcast-domain add-ports</code></td>
</tr>
<tr>
<td>Remove ports from a broadcast domain</td>
<td><code>network port broadcast-domain remove-ports</code></td>
</tr>
</tbody>
</table>

For more information about these commands, see the man pages.

Examples of adding and removing ports

The following command adds port e0g on node cluster-1-01 and port e0g on node cluster-1-02 to broadcast domain bcast1 in the “Default” IPspace:

```
cluster-1::> network port broadcast-domain add-ports -broadcast-domain bcast1 -ports cluster-1-01:e0g,cluster1-02:e0g
```

The following command adds two cluster ports to broadcast domain “Cluster” in the “Cluster” IPspace:

```
cluster-1::> network port broadcast-domain add-ports -broadcast-domain Cluster -ports cluster-2-03:e0f,cluster2-04:e0f -ipspace Cluster
```

The following command removes port e0e on node cluster1-01 from broadcast domain bcast1 in the “Default” IPspace:

```
cluster-1::> network port broadcast-domain remove-ports -broadcast-domain bcast1 -ports cluster-1-01:e0e
```

Related tasks

- [Displaying broadcast domains](#)
- [Creating a broadcast domain](#)
- [Splitting broadcast domains](#)
- [Merging broadcast domains](#)
Splitting broadcast domains

You can modify an existing broadcast domain by splitting it into two different broadcast domains, with each broadcast domain containing some of the original ports assigned to the original broadcast domain.

About this task

• If the ports are in a failover group, all of the ports in a failover group must be split.
• If the ports have LIFs associated with them, the LIFs cannot be part of a subnet's ranges.

Step

1. Split a broadcast domain into two broadcast domains:

   network port broadcast-domain split -ipspace ipspace_name -broadcast-domain broadcast-domain_name -new-broadcast-domain broadcast-domain_name -ports node:port,node:port

   • ipspace_name is the name of the ipspace where the broadcast domain resides.
   • -broadcast-domain is the name of the broadcast domain that will be split.
   • -new-broadcast-domain is the name of the new broadcast domain that will be created.
   • -ports is the node name and port to be added to the new broadcasts domain.

Related tasks

Adding or removing ports from a broadcast domain on page 50
Displaying information about failover groups on page 125
Displaying LIF information on page 121

Merging broadcast domains

You can move all of the ports from one broadcast domain into an existing broadcast domain using the merge command. This operation reduces the steps required if you were to remove all ports from a broadcast domain and then add the ports to an existing broadcast domain.

Step

1. Merge the ports from one broadcast domain into an existing broadcast domain:

   network port broadcast-domain merge -ipspace ipspace_name -broadcast-domain broadcast-domain_name -into-broadcast-domain broadcast-domain_name

   • ipspace_name is the name of the ipspace where the broadcast domains reside.
   • -broadcast-domain broadcast-domain_name is the name of the broadcast domain that will be merged.
   • -into-broadcast-domain broadcast-domain_name is the name of the broadcast domain that will receive additional ports.
   • -ports is the name of the ipspace where the broadcast domains reside.
-broadcast-domain broadcast-domain_name is the name of the broadcast domain that will be merged.

-into-broadcast-domain broadcast-domain_name is the name of the broadcast domain that will receive additional ports.

The following example merges broadcast domain bd-data1 into broadcast domain bd-data2:

```
network port -ipspace Default broadcast-domain bd-data1 into-broadcast-domain bd-data2
```

**Related tasks**

*Adding or removing ports from a broadcast domain* on page 50

*Splitting broadcast domains* on page 52

---

**Changing the MTU value for ports in a broadcast domain**

You can modify the MTU value for a broadcast domain to change the MTU value for all ports in that broadcast domain. This can be done to support topology changes that have been made in the network.

**Before you begin**

The MTU value must match all the devices connected to that layer 2 network.

**About this task**

Changing the MTU value causes a brief interruption in traffic over the affected ports. The system displays a prompt that you must answer with `y` to make the MTU change.

**Step**

1. Change the MTU value for all ports in a broadcast domain:

   ```
   network port broadcast-domain modify -broadcast-domain broadcast-domain_name -mtu mtu_value [ -ipspace ipspace_name ]
   ```

   - `broadcast-domain_name` is the name of the broadcast domain.
   - `mtu_value` is the MTU size for IP packets; 1500 and 9000 are typical values.
   - `ipspace_name` is the name of the IPspace in which this broadcast domain resides. The “Default” IPspace is used unless you specify a value for this option.

**Example**

The following command changes the MTU to 9000 for all ports in the broadcast domain bcast1:

```
cluster-1:~> network port broadcast-domain modify -broadcast-domain bcast1 -mtu 9000
Warning: Changing broadcast domain settings will cause a momentary data-serving
    interruption.
Do you want to continue? (y|n): y
```
Displaying broadcast domains

You can display the list of broadcast domains within each IPspace in a cluster. The output also shows the list of ports and the MTU value for each broadcast domain.

Step

1. Display the broadcast domains and associated ports in the cluster:

   \texttt{network port broadcast-domain show}

Example

The following command displays all of the broadcast domains and associated ports in the cluster:

\begin{verbatim}
cluster-1::> network port broadcast-domain show
IPspace Broadcast                                Update
Name    Domain Name  MTU   Port List             Status Details
------- ----------- -----  --------------------- --------------
Cluster Cluster  9000
cluster-1-01:e0a       complete
cluster-1-01:e0b       complete
cluster-1-02:e0a       complete
cluster-1-02:e0b       complete

Default Default      1500
cluster-1-01:e0c       complete
cluster-1-01:e0d       complete
cluster-1-02:e0c       complete
cluster-1-02:e0d       complete

bcast1       1500
cluster-1-01:e0e       complete
cluster-1-01:e0f       complete
cluster-1-01:e0g       complete
cluster-1-02:e0e       complete
cluster-1-02:e0f       complete
cluster-1-02:e0g       complete
\end{verbatim}

The following command displays the ports in the bcast1 broadcast domain that have an update status of \texttt{error}, which indicate that the port could not be updated properly:

\begin{verbatim}
cluster-1::> network port broadcast-domain show -broadcast-domain bcast1 -port-update-status error
IPspace Broadcast
Name    Domain Name  MTU   Port List             Status Details
------- ----------- -----  --------------------- --------------
Default bcast1      1500
cluster-1-02:e0g       error
\end{verbatim}

See the man page for additional options that you can use with this command.

Deleting a broadcast domain

If you no longer need a broadcast domain, you can delete it. This moves the ports associated with that broadcast domain to the “Default” IPspace.

Before you begin

There must be no subnets, network interfaces, or SVMs associated with the broadcast domain you want to delete.
About this task

- The system-created “Cluster” broadcast domain cannot be deleted.
- All failover groups related to the broadcast domain are removed when you delete the broadcast domain.

Step

1. Delete a broadcast domain:

   network port broadcast-domain delete -broadcast-domain
   broadcast_domain_name [-ipspace ipspace_name]

Example

The following command deletes broadcast domain bcast1 in IPspace ipspace1:

   cluster-1::> network port broadcast-domain delete -broadcast-domain
   bcast1 -ipspace ipspace1

Related tasks

   Displaying broadcast domains on page 54
Configuring failover groups and policies for LIFs

LIF failover refers to the automatic migration of a LIF to a different network port in response to a link failure on the LIF’s current port. This is a key component to providing high availability for the connections to SVMs. Configuring LIF failover involves creating a failover group, modifying the LIF to use the failover group, and specifying a failover policy.

A failover group contains a set of network ports (physical ports, VLANs, and interface groups) from one or more nodes in a cluster. The network ports that are present in the failover group define the failover targets available for the LIF. A failover group can have cluster management, node management, intercluster, and NAS data LIFs assigned to it.

Configuring LIF failover groups involves creating the failover group, modifying the LIF to use the failover group, and specifying a failover policy.

Attention: When a LIF is configured without a valid failover target, an outage occurs when the LIF attempts to fail over. You can use the network interface show -failover command to verify the failover configuration.

When you create a broadcast domain, a failover group of the same name is created automatically that contains the same network ports. This failover group is automatically managed by the system, meaning that as ports are added or removed from the broadcast domain, they are automatically added or removed from this failover group. This is provided as an efficiency for administrators who do not want to manage their own failover groups.

Related concepts

Configuring broadcast domains (cluster administrators only) on page 48

Related tasks

Creating a failover group on page 56
Configuring failover settings on a LIF on page 57

Creating a failover group

You create a failover group of network ports so that a LIF can automatically migrate to a different port if a link failure occurs on the LIF’s current port. This enables the system to reroute network traffic to other available ports in the cluster.

About this task

You use the network interface failover-groups create command to create the group and to add ports to the group.

• The ports added to a failover group can be network ports, VLANs, or interface groups (ifgrps).
• All of the ports added to the failover group must belong to the same broadcast domain.
• A single port can reside in multiple failover groups.
• If you have LIFs in different VLANs or broadcast domains, you must configure failover groups for each VLAN or broadcast domain.
• Failover groups do not apply in SAN iSCSI or FC environments.
Step

1. Create a failover group:

   network interface failover-groups create -vserver vserver_name -failover-group failover_group_name -targets ports_list

   - vserver_name is the name of the SVM that can use the failover group.
   - failover_group_name is the name of the failover group you want to create.
   - 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.

Example

The following command creates failover group fg3 for SVM vs3 and adds two ports:

   cluster-1::> network interface failover-groups create -vserver vs3 -failover-group fg3 -targets cluster1-01:e0e,cluster1-02:e0e

After you finish

- You should apply the failover group to a LIF now that the failover group has been created.
- Applying a failover group that does not provide a valid failover target for a LIF results in a warning message.
  If a LIF that does not have a valid failover target attempts to fail over, an outage might occur.

Related tasks

- Configuring failover settings on a LIF on page 57

Configuring failover settings on a LIF

You can configure a LIF to fail over to a specific group of network ports by applying a failover policy and a failover group to the LIF. You can also disable a LIF from failing over to another port.

About this task

- When a LIF is created, LIF failover is enabled by default, and the list of available target ports is determined by the default failover group and failover policy based on the LIF type and service policy.
  Starting with 9.5, you can specify a service policy for the LIF that defines which network services can use the LIF. Some network services impose failover restrictions on a LIF.

  **Note:** If a LIF's service policy is changed in a way that further restricts failover, the LIF's failover policy is automatically updated by the system.

- You can modify the failover behavior of LIFs by specifying values for the -failover-group and -failover-policy parameters in the network interface modify command.

- Modification of a LIF that results in the LIF having no valid failover target results in a warning message.
  If a LIF that does not have a valid failover target attempts to fail over, an outage might occur.

- The following list describes how the -failover-policy setting affects the target ports that are selected from the failover group:
  - broadcast-domain-wide
All ports on all nodes in the failover group.

- **system-defined**
  Only those ports on the LIF’s home node and one other node in the cluster, typically a non-SFO partner, if it exists.

- **local-only**
  Only those ports on the LIF’s home node.

- **sfo-partner-only**
  Only those ports on the LIF’s home node and its SFO partner.

- **disabled**
  The LIF is not configured for failover.

**Note:** Logical interfaces for SAN protocols do not support failover, therefore, these LIFs are always set to **disabled**.

**Step**

1. Configure failover settings for an existing interface:

   ```bash
   network interface modify -vserver vserver_name -lif lif_name -failover-policy failover_policy -failover-group failover_group
   ```

**Examples of configuring failover settings and disabling failover**

The following command sets the failover policy to **broadcast-domain-wide** and uses the ports in failover group fg3 as failover targets for LIF data1 on SVM vs3:

```
cluster-1::> network interface modify -vserver vs3 -lif data1 failover-policy broadcast-domain-wide -failover-group fg3
```

```
cluster-1::> network interface show -vserver vs3 -lif * -fields failover-group,failover-policy
vserver   lif             failover-policy        failover-group
-------   --------------  ---------------        --------------
vs3       data1           broadcast-domain-wide  fg3
```

The following command disables failover for LIF data1 on SVM vs3:

```
cluster-1::> network interface modify -vserver vs3 -lif data1 failover-policy disabled
```

**Related tasks**

- *Displaying information about failover groups* on page 125
- *Displaying LIF failover targets* on page 126
Commands for managing failover groups and policies

You can use the `network interface failover-groups` commands to manage failover groups. You use the `network interface modify` command to manage the failover groups and failover policies that are applied to a LIF.

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>Use this command...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add network ports to a failover group</td>
<td><code>network interface failover-groups add-targets</code></td>
</tr>
<tr>
<td>Remove network ports from a failover group</td>
<td><code>network interface failover-groups remove-targets</code></td>
</tr>
<tr>
<td>Modify network ports in a failover group</td>
<td><code>network interface failover-groups modify</code></td>
</tr>
<tr>
<td>Display the current failover groups</td>
<td><code>network interface failover-groups show</code></td>
</tr>
<tr>
<td>Configure failover on a LIF</td>
<td><code>network interface modify-failover-group-failover-policy</code></td>
</tr>
<tr>
<td>Display the failover group and failover policy that is being used by each LIF</td>
<td><code>network interface show -fields failover-group,failover-policy</code></td>
</tr>
<tr>
<td>Rename a failover group</td>
<td><code>network interface failover-groups rename</code></td>
</tr>
<tr>
<td>Delete a failover group</td>
<td><code>network interface failover-groups delete</code></td>
</tr>
</tbody>
</table>

**Note:** Modifying a failover group such that it does not provide a valid failover target for any LIF in the cluster can result in an outage when a LIF attempts to fail over.

For more information, see the man pages for the `network interface failover-groups` and `network interface modify` commands.

**Related tasks**

- *Creating a failover group* on page 56
- *Configuring failover settings on a LIF* on page 57
- *Displaying information about failover groups* on page 125
- *Displaying LIF failover targets* on page 126
Configuring subnets (cluster administrators only)

Subnets enable you to allocate specific blocks, or pools, of IP addresses for your ONTAP network configuration. This enables you to create LIFs more easily when using the `network_interface create` command, by specifying a subnet name instead of having to specify IP address and network mask values.

A subnet is created within a broadcast domain, and it contains a pool of IP addresses that belong to the same layer 3 subnet. IP addresses in a subnet are allocated to ports in the broadcast domain when LIFs are created. When LIFs are removed, the IP addresses are returned to the subnet pool and are available for future LIFs.

It is recommended that you use subnets because they make the management of IP addresses much easier, and they make the creation of LIFs a simpler process. Additionally, if you specify a gateway when defining a subnet, a default route to that gateway is added automatically to the SVM when a LIF is created using that subnet.

Related tasks

- Creating a subnet on page 60
- Adding or removing IP addresses from a subnet on page 61

Creating a subnet

You create a subnet to allocate, or reserve, specific blocks of IPv4 or IPv6 addresses for ONTAP network configuration. This enables you to create interfaces more easily by specifying a subnet name instead of having to specify the IP address and network mask values for each new interface.

Before you begin

The broadcast domain and IPspace where you plan to add the subnet must already exist.

About this task

- All subnet names must be unique within an IPspace.
- When adding IP address ranges to a subnet, you must ensure that there are no overlapping IP addresses in the network so that different subnets, or hosts, do not attempt to use the same IP address.
- If you specify a gateway when defining a subnet, a default route to that gateway is added automatically to the SVM when a LIF is created using that subnet. If you do not use subnets, or if you do not specify a gateway when defining a subnet, then you will need to use the `route create` command to add a route to the SVM manually.

Step

1. Create a subnet:

   ```bash
   network subnet create -subnet-name subnet_name -broadcast-domain broadcast_domain_name [-ipspace ipspace_name] -subnet subnet_address [-gateway gateway_address] [-ip-ranges ip_address_list] [-force-update-lif-associations true]
   ```

   - `subnet_name` is the name of the layer 3 subnet you want to create.
   - The name can be a text string like “Mgmt” or it can be a specific subnet IP value like 192.0.2.0/24.
• **broadcast_domain_name** is the name of the broadcast domain where the subnet will reside.

• **ipspace_name** is the name of the IPspace that the broadcast domain is part of.
  The “Default” IPspace is used unless you specify a value for this option.

• **subnet_address** is the IP address and mask of the subnet; for example, 192.0.2.0/24.

• **gateway_address** is the gateway for the default route of the subnet; for example, 192.0.2.1.

• **ip_address_list** is the list, or range, of IP addresses that will be allocated to the subnet.
  The IP addresses can be individual addresses, a range of IP addresses, or a combination in a comma-separated list.

• The value **true** can be set for the `-force-update-lif-associations` option.
  This command fails if any service processor or network interfaces are currently using the IP addresses in the specified range. Setting this value to **true** associates any manually addressed interfaces with the current subnet, and allows the command to succeed.

**Example**

The following command creates subnet sub1 in broadcast domain bcast1 in the “Default” IPspace. It adds an IPv4 subnet IP address and mask, the gateway, and a range of IP addresses:

```
cluster-1::> network subnet create -subnet-name sub1 -broadcast-domain bcast1 -subnet 192.0.2.0/24 -gateway 192.0.2.1 -ip-ranges 192.0.2.1-192.0.2.100, 192.0.2.122
```

The following command creates subnet sub2 in broadcast domain Default in the “Default” IPspace. It adds a range of IPv6 addresses:

```
cluster-1::> network subnet create -subnet-name sub2 -broadcast-domain Default -subnet 3FFE::/64 -gateway 3FFE::1 -ip-ranges "3FFE::10-3FFE::20"
```

**After you finish**

You can assign SVMs and interfaces to an IPspace using the addresses in the subnet.

If you need to change the name of an existing subnet, use the `network subnet rename` command.

**Related tasks**

- *Creating a LIF* on page 72
- *Adding or removing IP addresses from a subnet* on page 61
- *Changing subnet properties* on page 62

**Adding or removing IP addresses from a subnet**

You can add IP addresses when initially creating a subnet, or you can add IP addresses to a subnet that already exists. You can also remove IP addresses from an existing subnet. This enables you to allocate only the required IP addresses for SVMs.

**About this task**

When adding IP addresses, you will receive an error if any service processor or network interfaces are using the IP addresses in the range being added. If you want to associate any manually addressed interfaces with the current subnet, you can set the `-force-update-lif-associations` option to **true**.
When removing IP addresses, you will receive an error if any service processor or network interfaces are using the IP addresses being removed. If you want the interfaces to continue to use the IP addresses after they are removed from the subnet, you can set the `-force-update-lif-associations` option to `true`.

**Step**

1. Add or remove IP addresses from a subnet:

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>Use...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add IP addresses to a subnet</td>
<td><code>network subnet add-ranges</code></td>
</tr>
<tr>
<td>Remove IP addresses from a subnet</td>
<td><code>network subnet remove-ranges</code></td>
</tr>
</tbody>
</table>

For more information about these commands, see the man pages.

**Example**

The following command adds IP addresses 192.0.2.82 through 192.0.2.85 to subnet sub1:

```
cluster-1::> network subnet add-ranges -subnet-name sub1 -ip-ranges 192.0.2.82-192.0.2.85
```

The following command removes IP address 198.51.100.9 from subnet sub3:

```
cluster-1::> network subnet remove-ranges -subnet-name sub3 -ip-ranges 198.51.100.9
```

If the current range includes 1 through 10 and 20 through 40, and you want to add 11 through 19 and 41 through 50 (basically allowing 1 through 50), you can overlap the existing range of addresses by using the following command. This command adds only the new addresses and does not affect the existing addresses:

```
cluster-1::> network subnet add-ranges -subnet-name sub3 -ip-ranges 198.51.10.1-198.51.10.50
```

**Changing subnet properties**

You can change the subnet address and mask value, gateway address, or range of IP addresses in an existing subnet.

**About this task**

- When modifying IP addresses, you must ensure there are no overlapping IP addresses in the network so that different subnets, or hosts, do not attempt to use the same IP address.
- If you add or change the gateway IP address, the modified gateway is applied to new SVMs when a LIF is created in them using the subnet. A default route to the gateway is created for the SVM if the route does not already exist. You may need to manually add a new route to the SVM when you change the gateway IP address.

**Step**

1. Modify subnet properties:
**network subnet modify**

- **-subnet-name subnet_name**
- **[-ipspace ipspace_name]**
- **[-subnet subnet_address]**
- **[-gateway gateway_address]**
- **[-ip-ranges ip_address_list]**
- **[-force-update-lif-associations true]**

- **subnet_name** is the name of the subnet you want to modify.
- **ipspace_name** is the name of the IPspace where the subnet resides.
- **subnet_address** is the new address and mask of the subnet, if applicable; for example, 192.0.2.0/24.
- **gateway_address** is the new gateway of the subnet, if applicable; for example, 192.0.2.1. Entering "" removes the gateway entry.
- **ip_address_list** is the new list, or range, of IP addresses that will be allocated to the subnet, if applicable. The IP addresses can be individual addresses, a range or IP addresses, or a combination in a comma-separated list. The range specified here replaces the existing IP addresses.
- You can set the value **true** for the **-force-update-lif-associations** option when modifying the range of IP addresses. This command fails if any service processor or network interfaces are using the IP addresses in the specified range. Setting this value to **true** associates any manually addressed interfaces with the current subnet and allows the command to succeed.

**Example**

The following command modifies the gateway IP address of subnet sub3:

```
cluster-1::> network subnet modify -subnet-name sub3 -gateway 192.0.3.1
```

**Displaying subnets**

You can display the list of IP addresses that are allocated to each subnet within an IPspace. The output also shows the total number of IP addresses that are available in each subnet, and the number of addresses that are currently being used.

**Step**

1. Display the list of subnets and the associated IP address ranges that are used in those subnets:

   ```
   network subnet show
   ```

**Example**

The following command displays the subnets and the subnet properties:

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

<table>
<thead>
<tr>
<th>Subnet Name</th>
<th>Subnet Address</th>
<th>Broadcast Domain</th>
<th>Gateway Address</th>
<th>Avail/Total Ranges</th>
</tr>
</thead>
<tbody>
<tr>
<td>sub1</td>
<td>192.0.2.0/24</td>
<td>bcast1</td>
<td>192.0.2.1</td>
<td>5/9</td>
</tr>
<tr>
<td>sub3</td>
<td>198.51.100.0/24</td>
<td>bcast3</td>
<td>198.51.100.1</td>
<td>3/3</td>
</tr>
<tr>
<td></td>
<td>198.51.100.7,198.51.100.9</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Deleting a subnet

If you no longer need a subnet and want to deallocate the IP addresses that were assigned to the subnet, you can delete it.

**About this task**

You will receive an error if any service processor or network interfaces are currently using IP addresses in the specified ranges. If you want the interfaces to continue to use the IP addresses even after the subnet is deleted, you can set the `-force-update-lif-associations` option to `true` to remove the subnet's association with the LIFs.

**Step**

1. Delete a subnet:

   ```
   network subnet delete -subnet-name subnet_name [-ipspace ipspace_name] [-force-update-lif-associations true]
   ```

**Example**

The following command deletes subnet sub1 in IPspace ipspace1:

```
cluster-1::> network subnet delete -subnet-name sub1 -ipspace ipspace1
```
Configuring LIFs (cluster administrators only)

A LIF represents a network access point to a node in the cluster. You can configure LIFs on ports over which the cluster sends and receives communications over the network.

A cluster administrator can create, view, modify, migrate, or delete LIFs. An SVM administrator can only view the LIFs associated with the SVM.

Related tasks

Creating a LIF on page 72

Related references

Characteristics of LIFs on page 67

What LIFs are

A LIF (logical interface) is an IP address or WWPN with associated characteristics, such as a role, a home port, a home node, a list of ports to fail over to, and a firewall policy. You can configure LIFs on ports over which the cluster sends and receives communications over the network.

LIFs can be hosted on the following ports:

- Physical ports that are not part of interface groups
- Interface groups
- VLANs
- Physical ports or interface groups that host VLANs
- Virtual IP (VIP) ports

Starting with ONTAP 9.5, VIP LIFs are supported and are hosted on VIP ports.

While configuring SAN protocols such as FC on a LIF, it will be associated with a WWPN.

SAN administration

The following figure illustrates the port hierarchy in an ONTAP system:
Roles for LIFs

A LIF role determines the kind of traffic that is supported over the LIF, along with the failover rules that apply and the firewall restrictions that are in place. A LIF can have any one of the five roles: node management, cluster management, cluster, intercluster, and data.

node management LIF

A LIF that provides a dedicated IP address for managing a particular node in a cluster. Node management LIFs are created at the time of creating or joining the cluster. These LIFs are used for system maintenance, for example, when a node becomes inaccessible from the cluster.

cluster management LIF

A LIF that provides a single management interface for the entire cluster.

A cluster management LIF can fail over to any node management or data port in the cluster. It cannot fail over to cluster or intercluster ports.

cluster LIF

A LIF that is used to carry intracluster traffic between nodes in a cluster. Cluster LIFs must always be created on 10-GbE network ports.

Cluster LIFs can fail over between cluster ports on the same node, but they cannot be migrated or failed over to a remote node. When a new node joins a cluster, IP addresses are generated automatically. However, if you want to assign IP addresses manually to the cluster LIFs, you must ensure that the new IP addresses are in the same subnet range as the existing cluster LIFs.
data LIF
A LIF that is associated with a storage virtual machine (SVM) and is used for communicating with clients.

You can have multiple data LIFs on a port. These interfaces can migrate or fail over throughout the cluster. You can modify a data LIF to serve as an SVM management LIF by modifying its firewall policy to `mgmt`.

Sessions established to NIS, LDAP, Active Directory, WINS, and DNS servers use data LIFs.

intercluster LIF
A LIF that is used for cross-cluster communication, backup, and replication. You must create an intercluster LIF on each node in the cluster before a cluster peering relationship can be established.

These LIFs can only fail over to ports in the same node. They cannot be migrated or failed over to another node in the cluster.

Characteristics of LIFs
LIFs with different roles have different characteristics. A LIF role determines the kind of traffic that is supported over the interface, along with the failover rules that apply, the firewall restrictions that are in place, the security, the load balancing, and the routing behavior for each LIF.

LIF compatibility with port types
Note: When intercluster and management LIFs are configured in the same subnet to associate with a static route and if the route associates with an intercluster LIF, the management traffic will be blocked by an external firewall and the AutoSupport and NTP connections fail. You can recover the system by running the `network interface modify -vserver vserver name -lif intercluster LIF -status-admin up|down` command to toggle the intercluster LIF. However, you should set the intercluster LIF and management LIF in different subnets to avoid this issue.

<table>
<thead>
<tr>
<th></th>
<th>Data LIF</th>
<th>Cluster LIF</th>
<th>Node management LIF</th>
<th>Cluster management LIF</th>
<th>Intercluster LIF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary traffic types</td>
<td>NFS server, CIFS server, NIS client, Active Directory, LDAP, WINS, DNS client and server, iSCSI and FC server</td>
<td>Intracluster</td>
<td>SSH server, HTTPS server, NTP server, SNMP, AutoSupport client, DNS client, loading software updates</td>
<td>SSH server, HTTPS server</td>
<td>Cross-cluster replication</td>
</tr>
<tr>
<td>Notes</td>
<td>SAN LIFs cannot fail over. These LIFs also do not support load balancing.</td>
<td>Unauthenticated, unencrypted; essentially an internal Ethernet “bus” of the cluster.</td>
<td>Traffic flowing over intercluster LIFs is not encrypted.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### LIF security

<table>
<thead>
<tr>
<th>Require private IP subnet?</th>
<th>Data LIF</th>
<th>Cluster LIF</th>
<th>Node management LIF</th>
<th>Cluster management LIF</th>
<th>Intercluster LIF</th>
</tr>
</thead>
<tbody>
<tr>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Require secure network?</th>
<th>Data LIF</th>
<th>Cluster LIF</th>
<th>Node management LIF</th>
<th>Cluster management LIF</th>
<th>Intercluster LIF</th>
</tr>
</thead>
<tbody>
<tr>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Default firewall policy</th>
<th>Data LIF</th>
<th>Cluster LIF</th>
<th>Node management LIF</th>
<th>Cluster management LIF</th>
<th>Intercluster LIF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Very restrictive</td>
<td></td>
<td>Completely open</td>
<td>Medium</td>
<td>Medium</td>
<td>Very restrictive</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Is firewall customizable?</th>
<th>Data LIF</th>
<th>Cluster LIF</th>
<th>Node management LIF</th>
<th>Cluster management LIF</th>
<th>Intercluster LIF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

### LIF failover

<table>
<thead>
<tr>
<th>Default behavior</th>
<th>Data LIF</th>
<th>Cluster LIF</th>
<th>Node management LIF</th>
<th>Cluster management LIF</th>
<th>Intercluster LIF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Only those ports in the same failover group that are on the LIF’s home node and on a non-SFO partner node</td>
<td>Only those ports in the same failover group that are on the LIF’s home node</td>
<td>Only those ports in the same failover group that are on the LIF’s home node</td>
<td>Any port in the same failover group</td>
<td>Only those ports in the same failover group that are on the LIF’s home node</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Is customizable?</th>
<th>Data LIF</th>
<th>Cluster LIF</th>
<th>Node management LIF</th>
<th>Cluster management LIF</th>
<th>Intercluster LIF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

### LIF routing

<table>
<thead>
<tr>
<th>When is a default route needed?</th>
<th>Data LIF</th>
<th>Cluster LIF</th>
<th>Node management LIF</th>
<th>Cluster management LIF</th>
<th>Intercluster LIF</th>
</tr>
</thead>
<tbody>
<tr>
<td>When clients or domain controller are on different IP subnet</td>
<td>Never</td>
<td>When any of the primary traffic types require access to a different IP subnet</td>
<td>When administrator is connecting from another IP subnet</td>
<td>When other intercluster LIFs are on a different IP subnet</td>
<td></td>
</tr>
</tbody>
</table>
When is static route to a specific IP subnet needed?

- Never
- Rare
- Rare
- Rare

When nodes of another cluster have their intercluster LIFs in different IP subnets

When is static host route to a specific server needed?

- Never
- Rare
- Rare
- Rare

To have one of the traffic types listed under node management LIF go through a data LIF rather than a node management LIF. This requires a corresponding firewall change.

LIF rebalancing

<table>
<thead>
<tr>
<th>Data LIF</th>
<th>Cluster LIF</th>
<th>Node management LIF</th>
<th>Cluster management LIF</th>
<th>Intercluster LIF</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNS: use as DNS server?</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>DNS: export as zone?</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

Configuring LIF service policies

Starting with ONTAP 9.5, you can configure LIF service policies to identify a single service or a list of services that will use a LIF.

Creating a service policy for LIFs

Starting with ONTAP 9.5, you can create a service policy for LIFs. You can assign a service policy to one or more LIFs; thereby allowing the LIF to carry traffic for a single service or a list of services.

About this task

In ONTAP 9.5, service policies can only be used to configure a limited number of services. The following built-in service policies are available in the cluster:

- `net-intercluster`: Used by LIFs carrying intercluster traffic
• **net-route-announce**: Used by LIFs carrying BGP peer connections.

**Steps**

1. View the services that are available in the cluster:
   ```
   network interface service show
   ```

   Services represent the applications accessed by a LIF as well as the applications served by the cluster. Each service includes zero or more TCP and UDP ports on which the application is listening.

   In ONTAP 9.5, the following two services are available: **intercluster-core** (for core intercluster services) and **management-bgp** (for protocols related to BGP peer interactions).

   **Example**

   ```
   Service                Protocol:Port
   ---------------------- -------------
   intercluster-core      tcp:11104
   tcp:11105
   management-bgp         tcp:179
   2 entries were displayed.
   ```

2. At the advanced privilege level, create a service policy:
   ```
   network interface service-policy create -vserver svm_name -policy service_policy_name -services service_name -allowed-addresses IP_address/mask,...
   ```

   *service_name* specifies a list of services that should be included in the policy.

   *IP_address/mask* specifies the list of subnet masks for addresses that are allowed to access the services in the service policy. By default, all specified services are added with a default allowed address list of 0.0.0.0/0, which allows traffic from all subnets. When a non-default allowed address list is provided, LIFs using the policy are configured to block all requests with a source address that does not match any of the specified masks.

   **Example**

   ```
   cluster1::*> network interface service-policy create -vserver cluster1 -policy intercluster1 -services intercluster-core -allowed-addresses 10.1.0.0/16
   ```

3. Verify that the service policy is created:

   ```
   network interface service-policy show
   ```

   **Example**

   ```
   Vserver   Policy                 Service: Allowed Addresses
   --------- ---------------------- ----------------------------------
   cluster1 empty                  -
   intercluster1                  intercluster-core: 10.1.0.0/16
   net-intercluster               intercluster-core: 0.0.0.0/0
   3 entries were displayed.
   ```
Assign the service policy to a LIF either at the time of creation or by modifying an existing LIF.

Assigning a service policy to a LIF

Starting with ONTAP 9.5, you can assign a service policy to a LIF either at the time of creating the LIF or by modifying the LIF. A service policy defines the list of services that can be used with the LIF.

About this task

In ONTAP 9.5, the following built-in service policies are available in the cluster by default:

- `net-intercluster`: Used by LIFs carrying intercluster traffic
- `net-route-announce`: Used by LIFs carrying BGP peer connections

Step

1. Depending on when you want to assign the service policy to a LIF, perform one of the following actions:

<table>
<thead>
<tr>
<th>If you are...</th>
<th>Assign the service policy by entering the following command...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Creating a LIF</td>
<td>`network interface create -vserver svm_name -lif lif_name -home-node node_name -home-port port_name {(-address IP_address -netmask IP_address)</td>
</tr>
<tr>
<td>Modifying a LIF</td>
<td><code>network interface modify -vserver svm_name -lif lif_name -service-policy service_policy_name</code></td>
</tr>
</tbody>
</table>

When you specify a service policy for a LIF, you need not specify the data protocol and role for the LIF. Creating LIFs by specifying the role and data protocols is also supported.

Examples

The following example shows how to create a LIF and assign the `net-route-announce` service policy to a LIF:

```
network interface create -vserver julieccluster-2 -lif lif1 -address 10.53.39.208 -netmask 255.255.255.0 -service-policy net-route-announce -home-node juliec-vsim3 -home-port e0g
```

```
cluster1::> network interface create -vserver cluster1 -lif bgp_lif1 -home-node node-4 -home-port e1c -address 192.0.2.145 -netmask 255.255.255.0 -service-policy net-route-announce
```

The following example shows how to modify the service policy of an intercluster LIF to use the `net-route-announce` service policy:

```
cluster1::> network interface modify -vserver cluster1 -lif lif1 -service-policy net-route-announce
```
Commands for managing LIF service policies

Starting with ONTAP 9.5, you use the network interface service-policy commands to manage LIF service policies.

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>Use this command...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create a service policy</td>
<td>network interface service-policy create</td>
</tr>
<tr>
<td>Add an additional service entry to an existing service policy</td>
<td>network interface service-policy add-service</td>
</tr>
<tr>
<td>Clone an existing network service policy</td>
<td>network interface service-policy clone</td>
</tr>
<tr>
<td>Modify a service entry in an existing service policy</td>
<td>network interface service-policy modify-service</td>
</tr>
<tr>
<td>Remove a service entry from an existing service policy</td>
<td>network interface service-policy remove-service</td>
</tr>
<tr>
<td>Rename an existing network service policy</td>
<td>network interface service-policy rename</td>
</tr>
<tr>
<td>Delete an existing service policy</td>
<td>network interface service-policy delete</td>
</tr>
<tr>
<td>Restore a built-in service-policy to its original state</td>
<td>network interface service-policy restore-defaults</td>
</tr>
<tr>
<td>Display existing service policies</td>
<td>network interface service-policy show</td>
</tr>
</tbody>
</table>

Related information
ONTAP 9 commands

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 network.

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 cannot assign NAS and SAN protocols to the same LIF.
The supported protocols are CIFS, NFS, FlexCache, iSCSI, and FC; iSCSI and FC cannot be combined with other protocols. However, NAS and Ethernet-based SAN protocols can be present on the same physical port.

- You can create both IPv4 and IPv6 LIFs on the same network port.
- All the name mapping and host-name resolution services used by an SVM, such as DNS, NIS, LDAP, and Active Directory, must be reachable from at least one data LIF of the SVM.
- A cluster LIF should not be on the same subnet as a management LIF or a data LIF.
- Creating a LIF that does not have a valid failover target results in a warning message.
- 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 in 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 management LIF 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 data LIF can be configured per SVM

**Steps**

1. Create a LIF:

   ```
   network interface create -vserver vserver_name -lif lif_name -role lif_role -data-protocol {nfs|cifs|iscsi|fcp|fcache|none} -home-node node_name -home-port port_name { -address IP_address -netmask IP_address | -subnet-name subnet_name} -firewall-policy policy -auto-revert {true|false} -service-policy service_policy_name
   ```

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

- `--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.

- Starting with ONTAP 9.5, you can assign a service policy for the LIF with the `--service-policy` option.
  When a service policy is specified for a LIF, the policy is used to construct a default role, failover policy, and data protocol list for the LIF.

2. Optional: If you want to assign an IPv6 address in the `--address` option:
   a. Use the `network ndp prefix show` command to view the list of RA prefixes learned on various interfaces.
      The `network ndp prefix show` command is available at the advanced privilege level.
   b. Use the format `prefix::id` to construct the IPv6 address manually.
      For deriving the `id`, choose a random 64-bit hexadecimal number.

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

4. Verify that the configured IP address is reachable:

   | IPv4 address | network ping |
   | IPv6 address | network ping6 |

**Examples**

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

```bash
cluster-1::> network interface create -vserver vs1.example.com -lif datalif1 -role data -data-protocol cifs,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):

```bash
cluster-1::> network interface create -vserver vs3.example.com -lif datalif3 -role data -data-protocol cifs,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:

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

<table>
<thead>
<tr>
<th>Vserver</th>
<th>Logical Interface</th>
<th>Status</th>
<th>Network Address/Mask</th>
<th>Current Node</th>
<th>Current Is Port</th>
<th>Current Is Home</th>
</tr>
</thead>
<tbody>
<tr>
<td>cluster-1</td>
<td>cluster_mgmt</td>
<td>up/up</td>
<td>192.0.2.3/24</td>
<td>node-1</td>
<td>ela</td>
<td>true</td>
</tr>
<tr>
<td>node-1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```
<table>
<thead>
<tr>
<th>Node</th>
<th>Status</th>
<th>IP Address</th>
<th>Node Name</th>
<th>Interface</th>
<th>Failover Policy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clus 1</td>
<td>up/up</td>
<td>192.0.2.12/24</td>
<td>node-1</td>
<td>e0a</td>
<td>true</td>
</tr>
<tr>
<td>Clus 2</td>
<td>up/up</td>
<td>192.0.2.13/24</td>
<td>node-1</td>
<td>e0b</td>
<td>true</td>
</tr>
<tr>
<td>Mgmt 1</td>
<td>up/up</td>
<td>192.0.2.68/24</td>
<td>node-1</td>
<td>e1a</td>
<td>true</td>
</tr>
</tbody>
</table>

The following command shows how to create an intercluster LIF and assign a service policy, policy1, to it:

```
cluster1::> network interface create -vserver siteA -lif nodel_inter1 -service-policy policy1
```

### Related concepts
- Configuring failover groups and policies for LIFs on page 56
- Configuring subnets (cluster administrators only) on page 60

### Related tasks
- Configuring failover settings on a LIF on page 57
- Creating a failover group on page 56
- Migrating a LIF on page 76
- Reverting a LIF to its home port on page 78
- Creating a service policy for LIFs on page 69
- Configuring virtual IP (VIP) LIFs on page 79

### Related references
- Characteristics of LIFs on page 67

## Modifying a LIF

You can modify a LIF by changing the attributes, such as home node or current node, administrative status, IP address, netmask, failover policy, firewall policy, and service policy. You can also change the address family of a LIF from IPv4 to IPv6.

### About this task
- When modifying a LIF's administrative status to `down`, any outstanding NFSv4 locks are held until the LIF's administrative status is returned to `up`. To avoid lock conflicts that can occur when other LIFs attempt to access the locked files, you must move the NFSv4 clients to a different LIF before setting the administrative status to `down`.
- To modify a data LIF with NAS protocols to also serve as an SVM management LIF, you must modify the data LIF's firewall policy to `mgmt`.
- You cannot modify the data protocols used by a LIF. To modify the data protocols used by a LIF, you must delete and re-create the LIF.
- You cannot modify either the home node or the current node of a node management LIF.
You do not specify the home node when modifying the home port of a cluster LIF.

When using a subnet to change the IP address and network mask value for a LIF, an IP address is allocated from the specified subnet; if the LIF's previous IP address is from a different subnet, the IP address is returned to that subnet.

To modify the address family of a LIF from IPv4 to IPv6, you must use the colon notation for the IPv6 address and add a new value for the -netmask-length parameter.

You cannot modify the auto-configured link-local IPv6 addresses.

Modification of a LIF that results in the LIF having no valid failover target results in a warning message.

If a LIF that does not have a valid failover target attempts to fail over, an outage might occur.

Starting with ONTAP 9.5, you can modify the service policy associated with a LIF.

Steps

1. Modify a LIF's attributes by using the network interface modify command.

   Example

   The following example shows how to modify LIF datalif1 that is located on the SVM vs0. The LIF's IP address is changed to 172.19.8.1 and its network mask is changed to 255.255.0.0.

   ```
   cluster-1::> network interface modify -vserver vs0 -lif datalif1 -address 172.19.8.1 -netmask 255.255.0.0
   ```

   The following example shows how to modify the IP address and network mask of LIF datalif2 using an IP address and the network mask value from subnet client1_sub:

   ```
   cluster-1::> network interface modify -vserver vs1 -lif datalif2 -subnet-name client1_sub
   ```

   The following example shows how to modify the service policy of a LIF.

   ```
   clustershell::> network interface modify -vserver siteA -lif node1_inter1 -service-policy example
   ```

2. Verify that the IP addresses are reachable.

<table>
<thead>
<tr>
<th>If you are using</th>
<th>Then use</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPv4 addresses</td>
<td>network ping</td>
</tr>
<tr>
<td>IPv6 addresses</td>
<td>network ping6</td>
</tr>
</tbody>
</table>

   Related tasks

   * Creating a service policy for LIFs on page 69

Migrating a LIF

You might have to migrate a LIF to a different port on the same node or a different node within the cluster, if the port is either faulty or requires maintenance. Migrating a LIF is similar to LIF failover,
but LIF migration is a manual operation, while LIF failover is the automatic migration of a LIF in response to a link failure on the LIF's current network port.

Before you begin

- A failover group must have been configured for the LIFs.
- The destination node and ports must be operational and must be able to access the same network as the source port.

About this task

- You must migrate LIFs hosted on the ports belonging to a NIC to other ports in the cluster, before removing the NIC from the node.
- You must execute the command for migrating a cluster LIF from the node where the cluster LIF is hosted.
- A node-management LIF cannot be migrated to a remote node.
- When an NFSv4 LIF is migrated between nodes, a delay of up to 45 seconds results before the LIF is available on a new port.
  To work around this problem, use NFSv4.1 where no delay is encountered.
- You cannot migrate iSCSI LIFs from one node to another node.
  To work around this restriction, you must create an iSCSI LIF on the destination node. For information about guidelines for creating an iSCSI LIF, see the SAN Administration Guide.
- VMware VAAI copy offload operations fail when you migrate the source or the destination LIF.
  For more information about VMware VAAI, see the File Access and Protocols Management Guide.

Step

1. Depending on whether you want to migrate a specific LIF or all the LIFs, perform the appropriate action:

<table>
<thead>
<tr>
<th>If you want to migrate...</th>
<th>Enter the following command...</th>
</tr>
</thead>
<tbody>
<tr>
<td>A specific LIF</td>
<td><code>network interface migrate</code></td>
</tr>
<tr>
<td>All the data and cluster-management LIFs on a node</td>
<td><code>network interface migrate-all</code></td>
</tr>
<tr>
<td>All of the LIFs off of a port</td>
<td><code>network interface migrate-all -node &lt;node&gt; -port &lt;port&gt;</code></td>
</tr>
</tbody>
</table>

Example

The following example shows how to migrate a LIF named datalif1 on the SVM vs0 to the port e0d on node0b:

```
cluster-1:~> network interface migrate -vserver vs0 -lif datalif1 -dest-node node0b -dest-port e0d
```

The following example shows how to migrate all the data and cluster-management LIFs from the current (local) node:

```
cluster-1:~> network interface migrate-all -node local
```
Reverting a LIF to its home port

You can revert a LIF to its home port after it fails over or is migrated to a different port either manually or automatically. If the home port of a particular LIF is unavailable, the LIF remains at its current port and is not reverted.

About this task

• If you administratively bring the home port of a LIF to the up state before setting the automatic revert option, the LIF is not returned to the home port.

• The node management LIF does not automatically revert unless the value of the auto-revert option is set to true.

• You must ensure that the auto-revert option is enabled for the cluster LIFs to revert to their home ports.

Step

1. Revert a LIF to its home port manually or automatically:

   If you want to revert a LIF to its home port… Then enter the following command…
   Manually network interface revert -vserver vserver_name -lif lif_name
   Automatically network interface modify -vserver vserver_name -lif lif_name -auto-revert true

Related tasks
Displaying LIF information on page 121

Deleting a LIF

You can delete an LIF that is no longer required.

Before you begin
LIFs to be deleted must not be in use.

Steps

1. Use the network interface delete command to delete one or all LIFs:

   If you want to delete... Enter the command ...
   A specific LIF network interface delete -lif lif_name
   All LIFs network interface delete -lif *

Example

The following command deletes the LIF mgmtlif2:

cluster-1:/> network interface delete -vserver vs1 -lif mgmtlif2
2. Use the `network interface show` command to confirm that the LIF is deleted.

Related tasks

- Displaying LIF information on page 121

### Configuring virtual IP (VIP) LIFs

Some next-generation data centers use Network Layer 3 mechanisms that require LIFs to be failed over across subnets. Starting with ONTAP 9.5, VIP data LIFs and the associated routing protocol, border gateway protocol (BGP), are supported, which enable ONTAP to participate in these next-generation networks.

**About this task**

A VIP data LIF is a LIF that is not part of any subnet and is reachable from all ports that host a BGP LIF in the same IPspace. A VIP data LIF eliminates the dependency of a host on individual network interfaces. Because multiple physical adapters carry the data traffic, the entire load is not concentrated on a single adapter and the associated subnet. The existence of a VIP data LIF is advertised to peer routers through the routing protocol, Border Gateway Protocol (BGP).

VIP data LIFs provide the following advantages:

- **LIF portability beyond a broadcast domain or subnet**: VIP data LIFs can fail over to any subnet in the network by announcing the current location of each VIP data LIF to routers through BGP.
- **Aggregate throughput**: VIP data LIFs can support aggregate throughput that exceeds the bandwidth of any individual port because the VIP LIFs can send or receive data from multiple subnets or ports simultaneously.

### Setting up Border Gateway Protocol (BGP)

Starting with ONTAP 9.5, virtual IP (VIP) LIFs are supported. Before creating VIP LIFs, you must set up BGP, which is the routing protocol used for announcing the existence of VIP LIF to peer routers.

**Before you begin**

Peer router must be configured to accept BGP connection from the BGP LIF for the configured autonomous system number (ASN).

**Note**: ONTAP does not process any incoming route announcements from the router; therefore, you should configure the peer router for not sending any route updates to the cluster.

**About this task**

Setting up BGP involves optionally creating a BGP configuration, creating a BGP LIF, and creating a BGP peer group. ONTAP automatically creates a default BGP configuration with default values when the first BGP peer group is created on a given node. A BGP LIF is used to establish BGP TCP sessions with peer routers. For a peer router, a BGP LIF is the next hop to reach a VIP LIF. Failover is disabled for the BGP LIF. A BGP peer group advertises the VIP routes for all of the SVMs in the peer group’s IPspace.

**Steps**

1. Log in to the advanced privilege level:

   ```bash
   set -privilege advanced
   ```
2. Optional: Create a BGP configuration or modify the default BGP configuration of the cluster by performing one of the following actions:

- Create a BGP configuration:

  \[
  \text{network bgp config create -node \{node_name | local\} -asn \text{asn_integer} -holdtime \text{hold_time} -routerid \text{local_router_IP_address}}
  \]

  \[
  \text{cluster1::*> network bgp config create -node node1 -asn 65502 -holdtime 180 -routerid 1.1.1.1}
  \]

- Modify the default BGP configuration:

  \[
  \text{network bgp defaults modify -asn \text{asn_integer} -holdtime \text{hold_time}}
  \]

  \[
  \text{cluster1::*> network bgp defaults modify -asn 65502}
  \]

  \text{\textit{asn_integer} specifies the ASN. ASN for BGP is a non-negative 16-bit integer. The default ASN is 65501.}

  \text{\textit{hold_time} specifies the hold time in seconds. The default value is 180s.}

3. Create a BGP LIF for the system SVM:

  \[
  \text{network interface create -vserver \text{system_svm} -lif \text{lif_name} -service-policy net-route-announce -home-node \text{home_node} -home-port \text{home_port} -address \text{ip_address} -netmask \text{netmask}}
  \]

  You can use the \textit{net-route-announce} service policy for the BGP LIF.

  \textbf{Example}

  \[
  \text{cluster1::> network interface create -vserver cluster1 -lif bgp1 -service-policy net-route-announce -home-node cluster1-01 -home-port e0c -address 10.10.10.100 -netmask 255.255.255.0}
  \]

4. Create a BGP peer group that is used to establish BGP sessions with the remote peer routers and configure the VIP route information that is advertised to the peer routers.

a. Log in to the advance privilege level:

  \[
  \text{set -privilege advanced}
  \]

b. Create a BGP peer group to specify the BGP router that advertises the VIP LIFs in the specified data SVMs:

  \[
  \text{network bgp peer-group create -peer-group \text{group_name} -ipspace \text{ipspace_name} -local-lif \text{bgp_lif} -peer-address \text{peer-router_ip_address} -peer-asn \text{65502} -route-preference \text{integer}}
  \]

  \textbf{Example}

  \[
  \text{cluster1::> network bgp peer-group create -peer-group group1 -ipspace Default -local-lif bgp1 -peer-address 10.10.10.1 -peer-asn 65502 -route-preference 100}
  \]

\textbf{Related tasks}

\textit{Creating a static route} on page 101
Creating a virtual IP (VIP) data LIF

Starting with ONTAP 9.5, you can create a VIP data LIF. The existence of a VIP data LIF is advertised to peer routers through the routing protocol, Border Gateway Protocol (BGP).

Before you begin

- The BGP peer group must be set up and the BGP session for the SVM on which the LIF is to be created must be active.

  Setting up Border Gateway Protocol (BGP)

- A static route to the BGP router or any other router in the BGP LIF’s subnet must be created for any outgoing VIP traffic for the SVM.

- You should turn on multipath routing so that the outgoing VIP traffic can utilize all the available routes.

  Enabling multipath routing

  If multipath routing is not enabled, all the outgoing VIP traffic goes from a single interface.

Steps

1. Create a VIP data LIF:

   ```
   network interface create -vserver svm_name -lif lif_name -role data -data-protocol {nfs|cifs|fcache|none|fc-nvme} -home-node home_node -address ip_address -is-vip true
   ```

   A VIP port is automatically selected if you do not specify the home port with the `network interface create` command.

   By default, the VIP data LIF belongs to the system-created broadcast domain named ‘Vip’, for each IPSpace. You cannot modify the VIP broadcast domain.

   A VIP data LIF is reachable simultaneously on all ports hosting a BGP LIF of an IPSpace. If there is no active BGP session for the VIP’s SVM on the local node, the VIP data LIF fails over to the next VIP port on the node that has a BGP session established for that SVM.

Example

```bash
cluster1::*> network interface create -vserver vs34 -lif vip1 -role data -data-protocol cifs,nfs,fcache -home-node gw-node1 -address 3.3.3.3 -is-vip true
```

2. Verify that the BGP session is in the `up` status for the SVM of the VIP data LIF:

   ```
   network bgp vserver-status show
   ```

   If the BGP status is `down` for the SVM on a node, the VIP data LIF fails over to a different node where the BGP status is `up` for the SVM. If BGP status is `down` on all the nodes, the VIP data LIF cannot be hosted anywhere, and has LIF status as `down`.

Example

```bash
cluster1::> network bgp vserver-status show

Node    Vserver    bgp status
-------    --------    ---------
nodel    vs1        up
```
Commands for managing the Border Gateway Protocol (BGP)

Starting with ONTAP 9.5, you use the `network bgp` commands to manage the BGP sessions in ONTAP.

Managing BGP configuration

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>Use this command...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create a BGP configuration</td>
<td><code>network bgp config create</code></td>
</tr>
<tr>
<td>Modify BGP configuration</td>
<td><code>network bgp config modify</code></td>
</tr>
<tr>
<td>Delete BGP configuration</td>
<td><code>network bgp config delete</code></td>
</tr>
<tr>
<td>Display BGP configuration</td>
<td><code>network bgp config show</code></td>
</tr>
<tr>
<td>Displays the BGP status for the SVM of the VIP LIF</td>
<td><code>network bgp vserver-status show</code></td>
</tr>
</tbody>
</table>

Managing BGP default values

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>Use this command...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modify BGP default values</td>
<td><code>network bgp defaults modify</code></td>
</tr>
<tr>
<td>Display BGP default values</td>
<td><code>network bgp defaults show</code></td>
</tr>
</tbody>
</table>

Managing BGP peer groups

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>Use this command...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create a BGP peer group</td>
<td><code>network bgp peer-group create</code></td>
</tr>
<tr>
<td>Modify a BGP peer group</td>
<td><code>network bgp peer-group modify</code></td>
</tr>
<tr>
<td>Delete a BGP peer group</td>
<td><code>network bgp peer-group delete</code></td>
</tr>
<tr>
<td>Display BGP peer groups information</td>
<td><code>network bgp peer-group show</code></td>
</tr>
<tr>
<td>Rename a BGP peer group</td>
<td><code>network bgp peer-group rename</code></td>
</tr>
</tbody>
</table>

Related information

*ONTAP 9 commands*
Configuring host-name resolution

ONTAP must be able to translate host names to numerical IP addresses in order to provide access to clients and to access services. You must configure storage virtual machines (SVMs) to use local or external name services to resolve host information. ONTAP supports configuring an external DNS server or configuring the local hosts file for host name resolution.

When using an external DNS server, you can configure Dynamic DNS (DDNS), which automatically sends new or changed DNS information from your storage system to the DNS server. Without dynamic DNS updates, you must manually add DNS information (DNS name and IP address) to the identified DNS servers when a new system is brought online or when existing DNS information changes. This process is slow and error-prone. During disaster recovery, manual configuration can result in a long downtime.

Configuring DNS for host-name resolution

You use DNS to access either local or remote sources for host information. You must configure DNS to access one or both of these sources.

ONTAP must be able to look up host information to provide proper access to clients. You must configure name services to enable ONTAP to access local or external DNS services to obtain the host information.

ONTAP stores name service configuration information in a table that is the equivalent of the /etc/nsswitch.conf file on UNIX systems.

Configuring an SVM and data LIFs for host-name resolution using an external DNS server

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.

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:

   \texttt{vserver services name-service dns create \textendash vserver vserver\_name \textendash domains domain\_name \textendash name\_servers ip\_addresses \textendash 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.example.com -domains example.com -name-servers 192.0.2.201,192.0.2.202 -state enabled

**Note:** Starting in 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. Enable DNS on LIFs owned by the SVM:

<table>
<thead>
<tr>
<th>If you are:</th>
<th>Use this command:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modifying an existing LIF</td>
<td><code>network interface modify -lif lifname -dns-zone zone-name</code></td>
</tr>
<tr>
<td>Creating a new LIF</td>
<td><code>network interface create -lif lifname -dns-zone zone-name</code></td>
</tr>
</tbody>
</table>

**Example**

```
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
network interface modify -lif datalif1 -dns-zone zonename.whatever.com
```

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 starting in ONTAP 9.2.

**Example**

```
cluster-1::> vserver services name-service dns check -vserver vs1.example.com

<table>
<thead>
<tr>
<th>Vserver</th>
<th>Name Server</th>
<th>Status</th>
<th>Status Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>vs1.example.com</td>
<td>10.0.0.50</td>
<td>up</td>
<td>Response time (msec): 2</td>
</tr>
<tr>
<td>vs1.example.com</td>
<td>10.0.0.51</td>
<td>up</td>
<td>Response time (msec): 2</td>
</tr>
</tbody>
</table>
```

### Configuring the name service switch table for host-name resolution

You must configure the name service switch table correctly to enable ONTAP to consult local or external name service to retrieve host information.

**Before you begin**

You must have decided which name service to use for host mapping in your environment.

**Steps**

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

   ```bash
   vserver services name-service ns-switch create -vserver vserver_name -database database_name -source source_names
   ```

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

   ```bash
   vserver services name-service ns-switch show -vserver vserver_name
   ```
The following example creates an entry in the name service switch table for SVM vs1 to first use the local hosts file and then an external DNS server to resolve host names:

```
cluster-1::> vserver services name-service ns-switch create -vserver vs1 -database hosts -sources files,dns
```

Managing the hosts table (cluster administrators only)

A cluster administrator can add, modify, delete, and view the host name entries in the hosts table of the admin storage virtual machine (SVM). An SVM administrator can configure the host name entries only for the assigned SVM.

Commands for managing local host-name entries

You can use the `vserver services name-service dns hosts` command to create, modify, or delete DNS host table entries.

When you are creating or modifying the DNS host-name entries, you can specify multiple alias addresses separated by commas.

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>Use this command...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create a DNS host-name entry</td>
<td><code>vserver services name-service dns hosts create</code></td>
</tr>
<tr>
<td>Modify a DNS host-name entry</td>
<td><code>vserver services name-service dns hosts modify</code></td>
</tr>
<tr>
<td>Delete a DNS host-name entry</td>
<td><code>vserver services name-service dns hosts delete</code></td>
</tr>
</tbody>
</table>

For more information, see the man pages for the `vserver services name-service dns hosts` commands.

Related concepts

*Configuring host-name resolution* on page 83

Related information

*NFS management*
Balancing network loads to optimize user traffic (cluster administrators only)

You can configure your cluster to serve client requests from appropriately loaded LIFs. This results in a more balanced utilization of LIFs and ports, which in turn allows for better performance of the cluster.

What DNS load balancing is

DNS load balancing helps in selecting an appropriately loaded data LIF and balancing user network traffic across all available ports (physical, interface groups, and VLANs).

With DNS load balancing, LIFs are associated with the load balancing zone of an SVM. A site-wide DNS server is configured to forward all DNS requests and return the least-loaded LIF based on the network traffic and the availability of the port resources (CPU usage, throughput, open connections, and so on). DNS load balancing provides the following benefits:

- New client connections balanced across available resources.
- No manual intervention required for deciding which LIFs to use when mounting a particular SVM.
- DNS load balancing supports NFSv3, NFSv4, NFSv4.1, CIFS, SMB 2.0, SMB 2.1, and SMB 3.0.

How DNS load balancing works

Clients mount an SVM by specifying an IP address (associated with a LIF) or a host name (associated with multiple IP addresses). By default, LIFs are selected by the site-wide DNS server in a round-robin manner, which balances the workload across all LIFs.

Round-robin load balancing can result in overloading some LIFs, so you have the option of using a DNS load balancing zone that handles the host-name resolution in an SVM. Using a DNS load balancing zone, ensures better balance of the new client connections across available resources, leading to improved performance of the cluster.

A DNS load balancing zone is a DNS server inside the cluster that dynamically evaluates the load on all LIFs and returns an appropriately loaded LIF. In a load balancing zone, DNS assigns a weight (metric), based on the load, to each LIF.

Every LIF is assigned a weight based on its port load and CPU utilization of its home node. LIFs that are on less-loaded ports have a higher probability of being returned in a DNS query. Weights can also be manually assigned.

Creating a DNS load balancing zone

You can create a DNS load balancing zone to facilitate the dynamic selection of a LIF based on the load, that is, the number of clients mounted on a LIF. You can create a load balancing zone while creating a data LIF.

Before you begin

The DNS forwarder on the site-wide DNS server must be configured to forward all requests for the load balancing zone to the configured LIFs.
The knowledge base article How to set up DNS load balancing in Cluster-Mode on the NetApp Support Site contains more information about configuring DNS load balancing using conditional forwarding.

About this task

- Any data LIF can respond to DNS queries for a DNS load balancing zone name.
- A DNS load balancing zone must have a unique name in the cluster, and the zone name must meet the following requirements:
  - It should not exceed 256 characters.
  - It should include at least one period.
  - The first and the last character should not be a period or any other special character.
  - It cannot include any spaces between characters.
  - Each label in the DNS name should not exceed 63 characters.
    A label is the text appearing before or after the period. For example, the DNS zone named storage.company.com has three labels.

Step

1. Use the `network interface create` command with the `dns-zone` option to create a DNS load balancing zone.

   If the load balancing zone already exists, the LIF is added to it. For more information about the command, see the man pages.

Example

The following example demonstrates how to create a DNS load balancing zone named storage.company.com while creating the LIF lif1:

```
cluster-1::> network interface create -vserver vs0 -lif lif1 -role data -home-node node1 -home-port e0c -address 192.0.2.129 -netmask 255.255.255.128 -dns-zone storage.company.com
```

Related tasks

- Creating a LIF on page 72

Related information

- NetApp KB Article 1013801: How to set up DNS load balancing in Cluster-Mode

Adding or removing a LIF from a load balancing zone

You can add or remove a LIF from the DNS load balancing zone of a storage virtual machine (SVM). You can also remove all the LIFs simultaneously from a load balancing zone.

Before you begin

- All the LIFs in a load balancing zone should belong to the same SVM.
- A LIF can be a part of only one DNS load balancing zone.
- Failover groups for each subnet must have been set up, if the LIFs belong to different subnets.

**About this task**

A LIF that is in the administrative **down** status is temporarily removed from the DNS load balancing zone. When the LIF returns to the administrative **up** status, the LIF is automatically added to the DNS load balancing zone.

**Step**

1. Add a LIF to or remove a LIF from a load balancing zone:

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>Enter...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add a LIF</td>
<td><code>network interface modify -vserver vserver_name -lif lif_name -dns-zone zone_name</code></td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>cluster-1::&gt; network interface modify -vserver vs1 -lif data1 -dns-zone cifs.company.com</td>
</tr>
</tbody>
</table>

| Remove a single LIF | `network interface modify -vserver vserver_name -lif lif_name -dns-zone none` |
|                     | Example: |
|                     | cluster-1::> network interface modify -vserver vs1 -lif data1 -dns-zone none |

| Remove all LIFs     | `network interface modify -vserver vserver_name -lif * -dns-zone none` |
|                     | Example: |
|                     | cluster-1::> network interface modify -vserver vs0 -lif * -dns-zone none |

**Note:** You can remove an SVM from a load balancing zone by removing all the LIFs in the SVM from that zone.

**Related tasks**

*Modifying a LIF* on page 75
Configuring network security using Federal Information Processing Standards (FIPS)

ONTAP is compliant in the Federal Information Processing Standards (FIPS) 140-2 for all SSL connections. You can turn on and off SSL FIPS mode, set SSL protocols globally, and turn off any weak ciphers such as RC4 within ONTAP.

By default, SSL on ONTAP is set with FIPS compliance disabled and SSL protocol enabled with the following:

- TLSv1.2
- TLSv1.1
- TLSv1

When SSL FIPS mode is enabled, SSL communication from ONTAP to external client or server components outside of ONTAP will use FIPS compliant crypto for SSL.

Enabling FIPS

It is recommended that all secure users adjust their security configuration immediately after system installation or upgrade. When SSL FIPS mode is enabled, SSL communication from ONTAP to external client or server components outside of ONTAP will use FIPS compliant crypto for SSL.

About this task

The following settings are recommended to enable FIPS:

- FIPS: on
- SSL protocol = {TLSv1.2}
- SSL ciphers = {ALL:!LOW:!aNULL:!EXP:!eNULL:!RC4}

Steps

1. Change to advanced privilege level:
   
   ```
   set -privilege advanced
   ```

2. Enable FIPS:

   ```
   security config modify -interface SSL -is-fips-enabled true
   ```

3. When prompted to continue, enter y

4. One by one, manually reboot each node in the cluster.

   ```
   mycluster-1:/> security config modify -interface SSL -is-fips-enabled true
   Warning: This command will enable FIPS compliance and can potentially cause some non-compliant components to fail. MetroCluster and Vserver DR require FIPS to be enabled on both sites in order to be compatible. Do you want to continue? [y|n]: y
   Warning: When this command completes, reboot all nodes in the cluster. This is necessary to prevent components from failing due to an inconsistent security configuration state in the cluster. To avoid a service
   ```
Disabling FIPS

If you are still running an older system configuration and want to configure ONTAP with backward compatibility, you can turn on SSLv3 only when FIPS is disabled.

About this task

The following settings are recommended to disable FIPS:

- **FIPS = false**
- **SSL protocol = SSLv3**
- **SSL ciphers = ALL:!LOW:!aNULL:!EXP:!eNULL**

Steps

1. Change to advanced privilege level:
   ```
   set -privilege advanced
   ```
2. Disable FIPS by typing:
   ```
   security config modify -interface SSL -supported-protocols SSLv3
   ```
3. When prompted to continue, enter
   ```
   y
   ```
4. Manually reboot each node in the cluster.

```
florawcluster-1::*> security config modify -interface SSL -supported-protocols SSLv3
Warning: Enabling the SSLv3 protocol may reduce the security of the interface, and is not recommended.
Do you want to continue? [y|n]: y
```

```
florawcluster-1::*> security config show
```

### Viewing FIPS compliance status

You can see whether the entire cluster is running the current security configuration settings.

Steps

1. One by one, reboot each node in the cluster.
Do not reboot all cluster nodes simultaneously. A reboot is required to make sure that all applications in the cluster are running the new security configuration, and for all changes to FIPS on/off mode, Protocols, and Ciphers.

2. View the current compliance status:

    security config show

<table>
<thead>
<tr>
<th>Interface</th>
<th>FIPS Mode</th>
<th>Supported Protocols</th>
<th>Supported Ciphers</th>
<th>Config Ready</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSL</td>
<td>false</td>
<td>TLSv1_2, TLSv1_1, TLSv1</td>
<td>ALL:!LOW:!aNULL:</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>!EXP:!eNULL</td>
<td></td>
</tr>
</tbody>
</table>
Configuring IPv6 addresses

IPv6 increases the IP address size from 32 bits (in IPv4) to 128 bits. This larger address space provides expanded routing and addressing capabilities. You can create LIFs with IPv6 addresses.

The following are some of the advantages of the IPv6 protocol:

- Large address header
- Address auto-configuration
- Neighbor Discovery
- Path MTU discovery

Although most of the IPv6 features have been implemented in clustered Data ONTAP 8.2, you should familiarize yourself with the unsupported features of IPv6 as well. You can enable IPv6 on the cluster before configuring various networking components with IPv6 addresses.

For detailed explanations about various IPv6 address states, address auto-configuration, and the neighbor discovery features of IPv6, see the relevant RFCs.

Related information

- IPv6 addressing architecture (RFC4291)
- Neighbor Discovery for IP version 6 (RFC4861)
- IPv6 Stateless Address Configuration (RFC4862)

Enabling IPv6 on the cluster

You can enable IPv6 on the cluster and configure and manage various networking objects as IPv6-addressed objects.

Before you begin

All the nodes in the cluster must be running ONTAP.

About this task

Attention: If you ever decide to disable IPv6, you must contact technical support.

Steps

1. Use the network options ipv6 modify command to enable IPv6 on the cluster.

   Example

   `cluster-1:> network options ipv6 modify -enabled true`

2. Use the network options ipv6 show command to verify that IPv6 is enabled on the cluster.
Enabling or disabling RA processing

Enabling or disabling router-advertisement (RA) messages enables you to control the routing configuration for selecting the correct network interface to communicate with its neighbors.

**Before you begin**

IPv6 must be enabled on the cluster.

**About this task**

IPv6 RA processing is enabled by default when IPv6 is enabled on the cluster.

**Note:** In general, static route processing is preferred over RA processing. To disable RA processing, you must use the following command to set the RA processing to

```
false
```

**Step**

1. Use the `network options ipv6 modify` command to enable or disable IPv6 RA processing on the cluster.

**Example**

```
cluster-1::> network options ipv6 modify -is-ra-processing-enabled {true|false}
```
Configuring QoS marking (cluster administrators only)

Network Quality of Service (QoS) marking helps you to prioritize different traffic types based on the network conditions to effectively utilize the network resources. You can set the differentiated services code point (DSCP) value of the outgoing IP packets for the supported traffic types per IPspace.

DSCP marking for UC Compliance

Beginning with ONTAP 9, you can enable differentiated services code point (DSCP) marking on outgoing (egress) IP packet traffic for a given protocol with a default or user-provided DSCP code. DSCP marking is a mechanism for classifying and managing network traffic and is a component of Unified Capability (UC) compliance.

DSCP marking (also known as QoS marking or quality of service marking) is enabled by providing an IPspace, protocol, and DSCP value. The protocols on which DSCP marking can be applied are NFS, CIFS, iSCSI, SnapMirror, NDMP, FTP, HTTP/HTTPS, SSH, Telnet, and SNMP.

Modifying QoS marking values

You can modify the Quality of Service (QoS) marking values for different protocols, for each IPspace.

Before you begin

All nodes in the cluster must be running the same version of ONTAP 9.

Step

1. Modify QoS marking values by using the `network qos-marking modify` command.

   The `-ipspace` parameter specifies the IPspace for which the QoS marking entry is to be modified.

   The `-protocol` parameter specifies the protocol for which the QoS marking entry is to be modified. The `network qos-marking modify` man page describes the possible values of the protocol.

   The `-dscp` parameter specifies the Differentiated Services Code Point (DSCP) value. The possible values ranges from 0 through 63.

   The `-is-enabled` parameter is used to enable or disable the QoS marking for the specified protocol in the IPspace provided by the `-ipspace` parameter.

Example

The following command enables the QoS marking for the NFS protocol in default IPspace:

```
network qos-marking modify -ipspace Default -protocol NFS -is-enabled true
```

The following command sets the DSCP value to 20 for the NFS protocol in the default IPspace:

```
network qos-marking modify -ipspace Default -protocol NFS -dscp 20
```
Displaying QoS marking values

You can display the QoS marking values for different protocols, for each IPspace.

Step

1. Display QoS marking values by using the network qos-marking show command.

Example

The following command displays the QoS marking for all protocols in the default IPspace:

```
cluster-1::> network qos-marking show -ipspace Default
IPspace     Protocol           DSCP  Enabled?
----------------- ----------------- -----  --------
Default       CIFS                 10  false
              FTP                  48  false
              HTTP-admin          48  false
              HTTP-filesrv        10  false
              NDMP                10  false
              NFS                 10  true
              SNMP                48  false
              SSH                 48  false
              SnapMirror          10  false
              Telnet              48  false
              iSCSI               10  false
11 entries were displayed.
```
Configuring firewall service and policies for LIFs

Setting up a firewall enhances the security of the cluster and helps prevent unauthorized access to the storage system. By default, the firewall service allows remote systems access to a specific set of default services for data, management, and intercluster LIFs.

Firewall policies can be used to control access to management service protocols such as SSH, HTTP, HTTPS, Telnet, NTP, NDMP, NDMPS, RSH, DNS, or SNMP. Firewall policies cannot be set for data protocols such as NFS or CIFS.

You can manage firewall service and policies in the following ways:

• Enabling or disabling firewall service
• Displaying the current firewall service configuration
• Creating a new firewall policy with the specified policy name and network services
• Applying a firewall policy to a logical interface
• Creating a new firewall policy that is an exact copy of an existing policy.
  You can use this to make a policy with similar characteristics within the same SVM, or to copy the policy to a different SVM.
• Displaying information about firewall policies
• Modifying the IP addresses and netmasks that are used by a firewall policy
• Deleting a firewall policy that is not being used by a LIF

LIF roles and default firewall policies

LIF firewall policies are used to restrict access to the cluster over each LIF. You need to understand how the default firewall policy affects system access over each type of LIF, and how you can customize a firewall policy to increase or decrease security over a LIF.

When configuring a LIF using the `network interface create` or `network interface modify` command, the value specified for the `-firewall-policy` parameter determines the service protocols and IP addresses that are allowed access to the LIF.

In many cases you can accept the default firewall policy value. In other cases you might need to restrict access to certain IP addresses and certain management service protocols. The available management service protocols include SSH, HTTP, HTTPS, Telnet, NTP, NDMP, NDMPS, RSH, DNS, and SNMP.

The firewall policy for all cluster LIFs defaults to "" and cannot be modified.

The following table describes the default firewall policies that are assigned to each LIF, depending on their role, when you create the LIF:

<table>
<thead>
<tr>
<th>Firewall policy</th>
<th>Default service protocols</th>
<th>Default access</th>
<th>LIFs applied to</th>
</tr>
</thead>
<tbody>
<tr>
<td>mgmt</td>
<td>DNS, HTTP, HTTPS, NDMP, NDMPS, NTP, SNMP, SSH</td>
<td>Any address (0.0.0.0/0)</td>
<td>Cluster management, SVM management, and node management LIFs</td>
</tr>
<tr>
<td>intercluster</td>
<td>HTTPS, NDMP, NDMPS</td>
<td>Any address (0.0.0.0/0)</td>
<td>All intercluster LIFs</td>
</tr>
</tbody>
</table>
### Firewall policy

<table>
<thead>
<tr>
<th>Firewall policy</th>
<th>Default service protocols</th>
<th>Default access</th>
<th>LIFs applied to</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>DNS, NDMP, NDMPS, PORTMAP (starting with ONTAP 9.4)</td>
<td>Any address (0.0.0.0/0)</td>
<td>All data LIFs</td>
</tr>
</tbody>
</table>

### Portmap service is configurable in firewall in ONTAP 9.4

In ONTAP 9.3 and earlier, the portmap service (rpcbind) was always accessible on port 111 in network configurations that relied on the built-in ONTAP firewall rather than a third-party firewall. Starting in ONTAP 9.4, you can modify firewall policies to control whether the portmap service is accessible on particular LIFs.

The portmap service maps RPC services to the ports on which they listen. Depending on your configuration, you may be able to disallow access to the service on specific types of LIFs, typically management and intercluster LIFs. In some circumstances, you might even be able to disallow access on data LIFs.

#### What behavior you can expect

The 9.4 behavior is designed to provide a seamless transition on upgrade. If the portmap service is already being accessed over specific types of LIFs, it will continue to be accessible over those types of LIFs. As in previous ONTAP versions, you can specify the services accessible within the firewall in the firewall policy for the LIF type.

**Important:** All nodes in the cluster must be upgraded for the 9.4 behavior to take effect. Only inbound traffic is affected.

The new rules are as follows:

- On upgrade, ONTAP adds the portmap service to all existing firewall policies, default or custom.
- When you create a new cluster or new IPspace, ONTAP adds the portmap service only to the default data policy, not to the default management or intercluster policies.
- You can add the portmap service to default or custom policies as needed, and remove the service as needed.

#### How to add or remove the portmap service

To add the portmap service to an SVM or cluster firewall policy (make it accessible within the firewall), enter:

```
cluster_1::> system services firewall policy create -vserver SVM -policy mgmt|intercluster|data|custom -service portmap
```

To remove the portmap service from an SVM or cluster firewall policy (make it inaccessible within the firewall), enter:

```
cluster_1::> system services firewall policy delete -vserver SVM -policy -policy mgmt|intercluster|data|custom -service portmap
```

You can use the `network interface modify` command to apply the firewall policy to an existing LIF. For complete command syntax, see the man pages.
Creating a firewall policy and assigning it to a LIF

Default firewall policies are assigned to each LIF when you create the LIF. In many cases, the default firewall settings work well and you do not need to change them. If you want to change the network services or IP addresses that can access a LIF, you can create a custom firewall policy and assign it to the LIF.

About this task

- You cannot create a firewall policy with the policy name data, intercluster, cluster, or mgmt. These values are reserved for the system-defined firewall policies.
- You cannot set or modify a firewall policy for cluster LIFs. The firewall policy for cluster LIFs is set to 0.0.0.0/0 for all services types.
- If you need to modify or remove services, you must delete the existing firewall policy and create a new policy.
- If IPv6 is enabled on the cluster, you can create firewall policies with IPv6 addresses. After IPv6 is enabled, data and mgmt firewall policies include ::/0, the IPv6 wildcard, in their list of accepted addresses.
- When using OnCommand System Manager to configure data protection functionality across clusters, you must ensure that the intercluster LIF IP addresses are included in the allowed list, and that HTTPS service is allowed on both the intercluster LIFs and on your company-owned firewalls. By default, the intercluster firewall policy allows access from all IP addresses (0.0.0.0/0) and enables HTTPS, NDMP, and NDMPS services. If you modify this default policy, or if you create your own firewall policy for intercluster LIFs, you must add each intercluster LIF IP address to the allowed list and enable HTTPS service.

Steps

1. Create a firewall policy that will be available to the LIFs on a specific SVM:

   ```
   system services firewall policy create -vserver vserver_name -policy policy_name -service network_service -allow-list ip_address/mask
   ```

   You can use this command multiple times to add more than one network service and list of allowed IP addresses for each service in the firewall policy.

2. Verify that the policy was added correctly by using the system services firewall policy show command.

3. Apply the firewall policy to a LIF:

   ```
   network interface modify -vserver vserver_name -lif lif_name -firewall-policy policy_name
   ```

4. Verify that the policy was added correctly to the LIF by using the network interface show -fields firewall-policy command.

Example of creating a firewall policy and applying it to a LIF

The following command creates a firewall policy named data_http that enables HTTP and HTTPS protocol access from IP addresses on the 10.10 subnet, applies that policy to the LIF named data1 on SVM vs1, and then shows all of the firewall policies on the cluster:
cluster-1::> system services firewall policy create -vserver vs1 -policy data_http -service http -allow-list 10.10.0.0/16
cluster-1::> system services firewall policy create -vserver vs1 -policy data_http -service https -allow-list 10.10.0.0/16
cluster-1::> system services firewall policy show
Vserver Policy       Service    Allowed
------- ------------ ---------- -------------------
cluster-1
data        dns        0.0.0.0/0
                 ndmp       0.0.0.0/0
                 ndmps      0.0.0.0/0
intercluster    https      0.0.0.0/0
                 ndmp       0.0.0.0/0
                 ndmps      0.0.0.0/0
cluster-1
mgmt        dns        0.0.0.0/0
                 http       0.0.0.0/0
                 https      0.0.0.0/0
                 ndmp       0.0.0.0/0
                 ndmps      0.0.0.0/0
                 ntp        0.0.0.0/0
                 snmp       0.0.0.0/0
                 ssh        0.0.0.0/0
vs1         data_http    http       10.10.0.0/16
                 https      10.10.0.0/16
cluster-1::> network interface modify -vserver vs1 -lif data1 -firewall-policy data_http
cluster-1::> network interface show -fields firewall-policy
vserver  lif                  firewall-policy
-------  -------------------- ---------------
Cluster  node1_clus_1
Cluster  node1_clus_2
Cluster  node2_clus_1
Cluster  node2_clus_2
cluster-1 cluster_mgmt        mgmt
cluster-1 node1_mgmt1          mgmt
cluster-1 node2_mgmt1          mgmt
vs1      data1                data_http
vs3      data2                data

Related tasks

Modifying a LIF on page 75
Creating a LIF on page 72

Related references

Characteristics of LIFs on page 67
Commands for managing firewall service and policies

You can use the `system services firewall` commands to manage firewall service, the `system services firewall policy` commands to manage firewall policies, and the `network interface modify` command to manage firewall settings for LIFs.

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>Use this command...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable or disable firewall service</td>
<td><code>system services firewall modify</code></td>
</tr>
<tr>
<td>Display the current configuration for</td>
<td><code>system services firewall show</code></td>
</tr>
<tr>
<td>firewall service</td>
<td></td>
</tr>
<tr>
<td>Create a firewall policy or add a</td>
<td><code>system services firewall policy</code></td>
</tr>
<tr>
<td>service to an existing firewall policy</td>
<td><code>create</code></td>
</tr>
<tr>
<td>Apply a firewall policy to a LIF</td>
<td><code>network interface modify -lif</code></td>
</tr>
<tr>
<td></td>
<td><code>lifname -firewall-policy</code></td>
</tr>
<tr>
<td>Modify the IP addresses and netmasks</td>
<td><code>system services firewall policy</code></td>
</tr>
<tr>
<td>associated with a firewall policy</td>
<td><code>modify</code></td>
</tr>
<tr>
<td>Display information about firewall</td>
<td><code>system services firewall policy</code></td>
</tr>
<tr>
<td>policies</td>
<td><code>show</code></td>
</tr>
<tr>
<td>Create a new firewall policy that is</td>
<td><code>system services firewall policy</code></td>
</tr>
<tr>
<td>an exact copy of an existing policy</td>
<td><code>clone</code></td>
</tr>
<tr>
<td>Delete a firewall policy that is not</td>
<td><code>system services firewall policy</code></td>
</tr>
<tr>
<td>used by a LIF</td>
<td><code>delete</code></td>
</tr>
</tbody>
</table>

For more information, see the man pages for the `system services firewall`, `system services firewall policy`, and `network interface modify` commands.
Managing routing in an SVM

The routing table for an SVM determines the network path the SVM uses to communicate with a destination. It’s important to understand how routing tables work so that you can prevent network problems before they occur.

Routing rules are as follows:

- ONTAP routes traffic over the most specific available route.
- ONTAP routes traffic over a default gateway route (having 0 bits of netmask) as a last resort, when more specific routes are not available.

In the case of routes with the same destination, netmask, and metric, there is no guarantee that the system will use the same route after a reboot or after an upgrade. This is especially an issue if you have configured multiple default routes.

It is a best practice to configure one default route only for an SVM. To avoid disruption, you should ensure that the default route is able to reach any network address that is not reachable by a more specific route. For more information, see NetApp KB Article 1000317: Network access might be disrupted by incorrect routing configuration in clustered DataONTAP.

Creating a static route

You can create static routes within a storage virtual machine (SVM) to control how LIFs use the network for outbound traffic. When you create a route entry associated with an SVM, the route will be used by all LIFs that are owned by the specified SVM and that are on the same subnet as the gateway.

Step 1. Use the `network route create` command to create a route.

Example

```
cluster-1:> network route create -vserver vs0 -destination 0.0.0.0/0 -gateway 10.61.208.1
```

Enabling multipath routing

If multiple routes have the same metric for a destination, only one of the routes is picked for outgoing traffic. This leads to other routes being unutilized for sending outgoing traffic. Starting with ONTAP 9.5, you can enable multipath routing to load balance and utilize all the available routes.

Steps 1. Log in to the advanced privilege level:

   ```
   set -privilege advanced
   ```

2. Enable multipath routing:

   ```
   network options multipath-routing modify -is-enabled true
   ```

   Multipath routing is enabled for all nodes in the cluster.
Deleting a static route

You can delete an unneeded static route from a storage virtual machine (SVM).

Step

1. Use the `network route delete` command to delete a static route.

   For more information about this command, see the `network route` man page.

   Example

   The following example deletes a static route associated with SVM vs0 with a gateway of 10.63.0.1 and a destination IP address of 0.0.0.0/0:

   ```
   cluster-1::> network route delete -vserver vs0 -gateway 10.63.0.1 -destination 0.0.0.0/0
   ```

Displaying routing information

You can display information about the routing configuration for each SVM on your cluster. This can help you diagnose routing problems involving connectivity issues between client applications or services and a LIF on a node in the cluster.

Steps

1. Use the `network route show` command to display routes within one or more SVMs.

   Example

   The following example shows a route configured in the vs0 SVM:

   ```
   cluster-1::> network route show
   (network route show)
   Vserver Destination Gateway Metric
   --------------- -------------- ----------- -----
   vs0 0.0.0.0/0 172.17.178.1 20
   ```

2. Use the `network route show-lifs` command to display the association of routes and LIFs within one or more SVMs.

   Example

   The following example shows LIFs with routes owned by the vs0 SVM:

   ```
   cluster-1::> network route show-lifs
   (network route show-lifs)
   Vserver: vs0
   Destination Gateway Logical Interfaces
   ------------------ ---------- -------------------
3. Use the `network route active-entry show` command to display installed routes on one or more nodes, SVMs, subnets, or routes with specified destinations.

**Example**

The following example shows all installed routes on a specific SVM:

```
cluster-1:*> network route active-entry show -vserver Data0

Vserver: Data0  
Node: node-1    
Subnet Group: 0.0.0.0/0

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>Interface</th>
<th>Metric</th>
<th>Flags</th>
</tr>
</thead>
<tbody>
<tr>
<td>127.0.0.1</td>
<td>127.0.0.1</td>
<td>lo</td>
<td>10</td>
<td>UHS</td>
</tr>
<tr>
<td>127.0.10.1</td>
<td>127.0.20.1</td>
<td>losk</td>
<td>10</td>
<td>UHS</td>
</tr>
<tr>
<td>127.0.20.1</td>
<td>127.0.20.1</td>
<td>losk</td>
<td>10</td>
<td>UHS</td>
</tr>
</tbody>
</table>

Vserver: Data0  
Node: node-1    
Subnet Group: fd20:8b1e:b255:814e::/64

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>Interface</th>
<th>Metric</th>
<th>Flags</th>
</tr>
</thead>
<tbody>
<tr>
<td>default</td>
<td>fd20:8b1e:b255:814e::1</td>
<td>e0d</td>
<td>20</td>
<td>UGS</td>
</tr>
<tr>
<td>fd20:8b1e:b255:814e::/64</td>
<td>link#4</td>
<td>e0d</td>
<td>0</td>
<td>UC</td>
</tr>
</tbody>
</table>

Vserver: Data0  
Node: node-2    
Subnet Group: 0.0.0.0/0

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>Interface</th>
<th>Metric</th>
<th>Flags</th>
</tr>
</thead>
<tbody>
<tr>
<td>127.0.0.1</td>
<td>127.0.0.1</td>
<td>lo</td>
<td>10</td>
<td>UHS</td>
</tr>
</tbody>
</table>

Vserver: Data0  
Node: node-2    
Subnet Group: 0.0.0.0/0

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>Interface</th>
<th>Metric</th>
<th>Flags</th>
</tr>
</thead>
<tbody>
<tr>
<td>127.0.10.1</td>
<td>127.0.20.1</td>
<td>losk</td>
<td>10</td>
<td>UHS</td>
</tr>
<tr>
<td>127.0.20.1</td>
<td>127.0.20.1</td>
<td>losk</td>
<td>10</td>
<td>UHS</td>
</tr>
</tbody>
</table>

Vserver: Data0  
Node: node-2    
Subnet Group: fd20:8b1e:b255:814e::/64

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>Interface</th>
<th>Metric</th>
<th>Flags</th>
</tr>
</thead>
<tbody>
<tr>
<td>default</td>
<td>fd20:8b1e:b255:814e::1</td>
<td>e0d</td>
<td>20</td>
<td>UGS</td>
</tr>
<tr>
<td>fd20:8b1e:b255:814e::/64</td>
<td>link#4</td>
<td>e0d</td>
<td>0</td>
<td>UC</td>
</tr>
<tr>
<td>fd20:8b1e:b255:814e::1</td>
<td>link#4</td>
<td>e0d</td>
<td>0</td>
<td>UHL</td>
</tr>
</tbody>
</table>

11 entries were displayed.
```
Removing dynamic routes from routing tables

When ICMP redirects are received for IPv4 and IPv6, dynamic routes are added to the routing table. Beginning in ONTAP 9.3, by default, the dynamic routes are removed after 300 seconds. If you want to maintain dynamic routes for a different amount of time, you can change the time out value.

About this task

You can set the timeout value from 0 to 65,535 seconds. If you set the value to 0, the routes never expire. Removing dynamic routes prevents loss of connectivity caused by the persistence of invalid routes.

Steps

1. Display the current timeout value.
   - For IPv4:
     network tuning icmp show
   - For IPv6:
     network tuning icmp6 show

2. Modify the timeout value.
   - For IPv4:
     network tuning icmp modify -node node_name -redirect-timeout timeout_value
   - For IPv6:
     network tuning icmp6 modify -node node_name -redirect-v6-timeout timeout_value

3. Verify that the timeout value was modified correctly.
   - For IPv4:
     network tuning icmp show
   - For IPv6:
     network tuning icmp6 show
Managing SNMP on the cluster (cluster administrators only)

You can configure SNMP to monitor SVMs in your cluster to avoid issues before they occur, and to respond to issues if they do occur. Managing SNMP involves configuring SNMP users and configuring SNMP traphost destinations (management workstations) for all SNMP events. SNMP is disabled by default on data LIFs.

You can create and manage read-only SNMP users in the data SVM. Data LIFs must be configured to receive SNMP requests on the SVM.

SNMP network management workstations, or managers, can query the SVM SNMP agent for information. The SNMP agent gathers information and forwards it to the SNMP managers. The SNMP agent also generates trap notifications whenever specific events occur. The SNMP agent on the SVM has read-only privileges; it cannot be used for any set operations or for taking a corrective action in response to a trap. ONTAP provides an SNMP agent compatible with SNMP versions v1, v2c, and v3. SNMPv3 offers advanced security by using passphrases and encryption.

For more information about SNMP support in ONTAP systems, see TR-4220 on the NetApp Support site.

mysupport.netapp.com

Related tasks

Creating a LIF on page 72

What MIBs are

A MIB (Management Information Base) is a text file that describes SNMP objects and traps. MIBs describe the structure of the management data of the storage system and they use a hierarchical namespace containing object identifiers (OIDs). Each OID identifies a variable that can be read by using SNMP.

Because MIBs are not configuration files and ONTAP does not read these files, SNMP functionality is not affected by MIBs. ONTAP provides the following MIB file:

- A NetApp custom MIB (`netapp.mib`)

ONTAP supports IPv6 (RFC 2465), TCP (RFC 4022), UDP (RFC 4113), and ICMP (RFC 2466) MIBs, which show both IPv4 and IPv6 data, are supported.

ONTAP also provides a short cross-reference between object identifiers (OIDs) and object short names in the `traps.dat` file.

Note: The latest versions of the ONTAP MIBs and `traps.dat` files are available on the NetApp Support Site. However, the versions of these files on the support site do not necessarily correspond to the SNMP capabilities of your ONTAP version. These files are provided to help you evaluate SNMP features in the latest ONTAP version.

Related information

NetApp Support Site: mysupport.netapp.com
SNMP traps

SNMP traps capture system monitoring information that is sent as an asynchronous notification from the SNMP agent to the SNMP manager. There are three types of SNMP traps: standard, built-in, and user-defined. User-defined traps are not supported in ONTAP.

A trap can be used to check periodically for operational thresholds or failures that are defined in the MIB. If a threshold is reached or a failure is detected, the SNMP agent sends a message (trap) to the traphosts alerting them of the event.

Note: ONTAP supports SNMPv1 traps and does not support SNMPv2c and SNMPv3 traps, and INFORMs.

Standard SNMP traps

These traps are defined in RFC 1215. There are five standard SNMP traps that are supported by ONTAP: coldStart, warmStart, linkDown, linkUp, and authenticationFailure.

Note: The authenticationFailure trap is disabled by default. You must use the system snmp authtrap command to enable the trap. See the man pages for more information.

Built-in SNMP traps

Built-in traps are predefined in ONTAP and are automatically sent to the network management stations on the traphost list if an event occurs. These traps, such as diskFailedShutdown, cpuTooBusy, and volumeNearlyFull, are defined in the custom MIB.

Each built-in trap is identified by a unique trap code.

Creating an SNMP community and assigning it to a LIF

You can create an SNMP community that acts as an authentication mechanism between the management station and the storage virtual machine (SVM) when using SNMPv1 and SNMPv2c. By creating SNMP communities in a data SVM, you can execute commands such as snmpwalk and snmpget on the data LIFs.

About this task

- In new installations of ONTAP, SNMPv1 and SNMPv2c are disabled by default. SNMPv1 and SNMPv2c are enabled after you create an SNMP community.
- ONTAP supports read-only communities.
- By default, the “data” firewall policy that is assigned to data LIFs has SNMP service set to deny. You must create a new firewall policy with SNMP service set to allow when creating an SNMP user for a data SVM.
- You can create SNMP communities for SNMPv1 and SNMPv2c users for both the admin SVM and the data SVM.
- Because an SVM is not part of the SNMP standard, queries on data LIFs must include the NetApp root OID (1.3.6.1.4.1.789)—for example, snmpwalk -v 2c -c snmpNFS 10.238.19.14 1.3.6.1.4.1.789.

Steps

1. Create an SNMP community by using the system snmp community add command.
Example
The following command shows how to create an SNMP community in the admin SVM cluster-1:

```
cluster-1::> system snmp community add -type ro -community-name comty1 -vserver cluster-1
```

The following command shows how to create an SNMP community in the data SVM vs1:

```
cluster-1::> system snmp community add -type ro -community-name comty2 -vserver vs1
```

2. Verify that the communities have been created by using the `system snmp community show` command.

Example
The following command shows the two communities created for SNMPv1 and SNMPv2c:

```
cluster-1::> system snmp community show
cluster-1
  ro  comty1
vs1
  ro  comty2
```

3. Check whether SNMP is allowed as a service in the “data” firewall policy by using the `system services firewall policy show` command.

Example
The following command shows that the `snmp` service is not allowed in the default “data” firewall policy (the `snmp` service is allowed in the “mgmt” firewall policy only):

```
cluster-1::> system services firewall policy show

<table>
<thead>
<tr>
<th>Vserver Policy</th>
<th>Service</th>
<th>Allowed</th>
</tr>
</thead>
<tbody>
<tr>
<td>cluster-1</td>
<td>dns</td>
<td>0.0.0.0/0</td>
</tr>
<tr>
<td>cluster-1</td>
<td>ndmp</td>
<td>0.0.0.0/0</td>
</tr>
<tr>
<td>cluster-1</td>
<td>ndmps</td>
<td>0.0.0.0/0</td>
</tr>
<tr>
<td>cluster-1</td>
<td>https</td>
<td>0.0.0.0/0</td>
</tr>
<tr>
<td>cluster-1</td>
<td>ndmp</td>
<td>0.0.0.0/0</td>
</tr>
<tr>
<td>cluster-1</td>
<td>ndmps</td>
<td>0.0.0.0/0</td>
</tr>
<tr>
<td>cluster-1</td>
<td>ntp</td>
<td>0.0.0.0/0</td>
</tr>
<tr>
<td>cluster-1</td>
<td>snmp</td>
<td>0.0.0.0/0</td>
</tr>
<tr>
<td>cluster-1</td>
<td>ssh</td>
<td>0.0.0.0/0</td>
</tr>
</tbody>
</table>
```

4. Create a new firewall policy that allows access using the `snmp` service by using the `system services firewall policy create` command.
Example

The following commands create a new data firewall policy named “data1” that allows the snmp service from any IP address, and verify that the policy has been created successfully:

```
cluster-1::> system services firewall policy create -policy data1 -service snmp -vserver vs1 -allow-list 0.0.0.0/0

cluster-1::> system services firewall policy show -service snmp
Vserver  Policy       Service    Allowed
-------  ------------ ---------- -------------------
cluster-1 mgmt    snmp       0.0.0.0/0
vs1       data1    snmp       0.0.0.0/0
```

5. Apply the firewall policy to a data LIF by using the `network interface modify` command with the `-firewall-policy` parameter.

Example

The following command assigns the new “data1” firewall policy to LIF datalif1:

```
cluster-1::> network interface modify -vserver vs1 -lif datalif1 -firewall-policy data1
```

Related tasks

*Creating a firewall policy and assigning it to a LIF* on page 98

Configuring SNMPv3 users in a cluster

SNMPv3 is a secure protocol when compared to SNMPv1 and SNMPv2c. To use SNMPv3, you must configure an SNMPv3 user to run the SNMP utilities from the SNMP manager.

**Step**

1. Use the `security login create` command to create an SNMPv3 user.

   You are prompted to provide the following information:

   - Engine ID: Default and recommended value is `local EngineID`
   - Authentication protocol
   - Authentication password
   - Privacy protocol
   - Privacy protocol password

**Result**

The SNMPv3 user can log in from the SNMP manager by using the user name and password and run the SNMP utility commands.
SNMPv3 security parameters

SNMPv3 includes an authentication feature that, when selected, requires users to enter their names, an authentication protocol, an authentication key, and their desired security level when invoking a command.

The following table lists the SNMPv3 security parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Command-line option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>engineID</td>
<td>-e EngineID</td>
<td>Engine ID of the SNMP agent. Default value is local EngineID (recommended).</td>
</tr>
<tr>
<td>securityName</td>
<td>-u Name</td>
<td>User name must not exceed 32 characters.</td>
</tr>
<tr>
<td>authProtocol</td>
<td>-a {none</td>
<td>MD5</td>
</tr>
<tr>
<td>authKey</td>
<td>-A PASSPHRASE</td>
<td>Passphrase with a minimum of eight characters.</td>
</tr>
<tr>
<td>securityLevel</td>
<td>-l {authNoPriv</td>
<td>AuthPriv</td>
</tr>
<tr>
<td>privProtocol</td>
<td>-x { none</td>
<td>des</td>
</tr>
<tr>
<td>privPassword</td>
<td>-X password</td>
<td>Password with a minimum of eight characters.</td>
</tr>
</tbody>
</table>

Examples for different security levels

This example shows how an SNMPv3 user created with different security levels can use the SNMP client-side commands, such as `snmpwalk`, to query the cluster objects.

For better performance, you should retrieve all objects in a table rather than a single object or a few objects from the table.

**Note:** You must use `snmpwalk` 5.3.1 or later when the authentication protocol is SHA.

Security level: authPriv

The following output shows the creation of an SNMPv3 user with the `authPriv` security level.

```
cluster-1:~> security login create -username snmpv3user -application snmp -authmethod usm
Please enter the authoritative entity's EngineID [local EngineID]:
Please choose an authentication protocol (none, md5, sha) [none]:sha
Please enter authentication protocol password (minimum 8 characters long):
Please enter authentication protocol password again:
Please choose a privacy protocol (none, des) [none]: des
Please enter privacy protocol password (minimum 8 characters long):
Please enter privacy protocol password again:
```
The following output shows the SNMPv3 user running the `snmpwalk` command:

```bash
$ snmpwalk -v 3 -u snmpv3user -a SHA -A password1! -x DES -X password1! -l authPriv 192.0.2.62 .1.3.6.1.4.1.789.1.5.8.1.2
enterprises.789.1.5.8.1.2.1028 = "vo10"
enterprises.789.1.5.8.1.2.1032 = "vo10"
enterprises.789.1.5.8.1.2.1038 = "root_vs0"
enterprises.789.1.5.8.1.2.1042 = "root_vstrap"
enterprises.789.1.5.8.1.2.1064 = "vol1"
```

**Security level: authNoPriv**

The following output shows the creation of an SNMPv3 user with the `authNoPriv` security level.

```bash
cluster-1::> security login create -username snmpv3user1 -application snmp -authmethod usm -role admin
Please enter the authoritative entity's EngineID [local EngineID]:
Please choose an authentication protocol (none, md5, sha) [none]: md5
Please enter authentication protocol password (minimum 8 characters long):
Please enter authentication protocol password again:
Please choose a privacy protocol (none, des) [none]: none
```

The following output shows the SNMPv3 user running the `snmpwalk` command:

```bash
$ snmpwalk -v 3 -u snmpv3user1 -a MD5 -A password1! -l authNoPriv 192.0.2.62 .1.3.6.1.4.1.789.1.5.8.1.2
enterprises.789.1.5.8.1.2.1028 = "vo10"
enterprises.789.1.5.8.1.2.1032 = "vo10"
enterprises.789.1.5.8.1.2.1038 = "root_vs0"
enterprises.789.1.5.8.1.2.1042 = "root_vstrap"
enterprises.789.1.5.8.1.2.1064 = "vol1"
```

**Security level: noAuthNoPriv**

The following output shows the creation of an SNMPv3 user with the `noAuthNoPriv` security level.

```bash
cluster-1::> security login create -username snmpv3user2 -application snmp -authmethod usm -role admin
Please enter the authoritative entity's EngineID [local EngineID]:
Please choose an authentication protocol (none, md5, sha) [none]: none
```

The following output shows the SNMPv3 user running the `snmpwalk` command:

```bash
$ snmpwalk -v 3 -u snmpv3user2 -l noAuthNoPriv 192.0.2.62 .1.3.6.1.4.1.789.1.5.8.1.2
enterprises.789.1.5.8.1.2.1028 = "vo10"
enterprises.789.1.5.8.1.2.1032 = "vo10"
enterprises.789.1.5.8.1.2.1038 = "root_vs0"
enterprises.789.1.5.8.1.2.1042 = "root_vstrap"
enterprises.789.1.5.8.1.2.1064 = "vol1"
```
Configuring traphosts to receive SNMP notifications

You can configure the traphost (SNMP manager) to receive notifications (SNMP trap PDUs) when SNMP traps are generated in the cluster. You can specify either the host name or the IP address (IPv4 or IPv6) of the SNMP traphost.

Before you begin

- SNMP and SNMP traps must be enabled on the cluster.
  
  Note: SNMP and SNMP traps are enabled by default.

- DNS must be configured on the cluster for resolving the traphost names.

- IPv6 must be enabled on the cluster to configure SNMP traphosts by using IPv6 addresses.

- For ONTAP 9.1 and later versions, you must have specified the authentication of a predefined User-based Security Model (USM) and privacy credentials when creating traphosts.

Step

1. Add an SNMP traphost:

   `system snmp traphost add`

   Note: Traps can be sent only when at least one SNMP management station is specified as a traphost.

Example

The following command adds a new SNMPv3 traphost named yyy.example.com with a known USM user:

```
cluster-1::> system snmp traphost add -peer-address yyy.example.com -usm-username MyUsmUser
```

Example

The following command adds a traphost using the IPv6 address of the host:

```
cluster-1::> system snmp traphost add -peer-address 2001:0db8:1::209:6bff:feae:6d67
```
Commands for managing SNMP

You can use the `system snmp` commands to manage SNMP, traps, and traphosts. You can use the `security` commands to manage SNMP users per SVM. You can use the `event` commands to manage events related to SNMP traps.

**Commands for configuring SNMP**

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>Use this command...</th>
</tr>
</thead>
</table>
| Enable SNMP on the cluster | `options -option-name snmp.enable -option-value on`  
  **Note:** The SNMP service must be allowed under the management (mgmt) firewall policy. You can verify whether SNMP is allowed by using the `system services firewall policy show` command. |
| Disable SNMP on the cluster | `options -option-name snmp.enable -option-value off` |

**Commands for managing SNMP v1, v2c, and v3 users**

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>Use this command...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configure SNMP users</td>
<td><code>security login create</code></td>
</tr>
<tr>
<td>Display SNMP users</td>
<td><code>security snmpusers</code> and <code>security login show -application snmp</code></td>
</tr>
<tr>
<td>Delete SNMP users</td>
<td><code>security login delete</code></td>
</tr>
<tr>
<td>Modify the access-control role name of a login method for SNMP users</td>
<td><code>security login modify</code></td>
</tr>
</tbody>
</table>

**Commands for providing contact and location information**

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>Use this command...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display or modify the contact details of the cluster</td>
<td><code>system snmp contact</code></td>
</tr>
<tr>
<td>Display or modify the location details of the cluster</td>
<td><code>system snmp location</code></td>
</tr>
</tbody>
</table>

**Commands for managing SNMP communities**

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>Use this command...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add a read-only (<strong>ro</strong>) community for an SVM or for all SVMs in the cluster</td>
<td><code>system snmp community add</code></td>
</tr>
<tr>
<td>Delete a community or all communities</td>
<td><code>system snmp community delete</code></td>
</tr>
<tr>
<td>Display the list of all communities</td>
<td><code>system snmp community show</code></td>
</tr>
</tbody>
</table>
Because SVMs are not part of the SNMP standard, queries on data LIFs must include the NetApp root OID (1.3.6.1.4.1.789), for example, `snmpwalk -v 2c -c snmpNFS 10.238.19.14 1.3.6.1.4.1.789`.

**Command for displaying SNMP option values**

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>Use this command...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display the current values of all SNMP options, including cluster contact, contact location, whether the cluster is configured to send traps, the list of traphosts, and list of communities and access control type</td>
<td><code>system snmp show</code></td>
</tr>
</tbody>
</table>

**Commands for managing SNMP traps and traphosts**

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>Use this command...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable SNMP traps sent from the cluster</td>
<td><code>system snmp init -init 1</code></td>
</tr>
<tr>
<td>Disable SNMP traps sent from the cluster</td>
<td><code>system snmp init -init 0</code></td>
</tr>
<tr>
<td>Add a traphost that receives SNMP notifications for specific events in the cluster</td>
<td><code>system snmp traphost add</code></td>
</tr>
<tr>
<td>Delete a traphost</td>
<td><code>system snmp traphost delete</code></td>
</tr>
<tr>
<td>Display the list of traphosts</td>
<td><code>system snmp traphost show</code></td>
</tr>
</tbody>
</table>

**Commands for managing events related to SNMP traps**

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>Use this command...</th>
</tr>
</thead>
</table>
| Display the events for which SNMP traps (built-in) are generated | `event route show`  
  - Use the `-snmp-support true` parameter to view only SNMP-related events.  
  - Use the `instance -messagename <message>` parameter to view a detailed description why an event might have occurred, and any corrective action.  
  **Note:** Routing of individual SNMP trap events to specific traphost destinations is not supported. All SNMP trap events are sent to all traphost destinations. |
| Display a list of SNMP trap history records, which are event notifications that have been sent to SNMP traps | `event snmphistory show` |
| Delete an SNMP trap history record | `event snmphistory delete` |

For more information about the `system snmp`, `security`, and `event commands`, see the man pages.
ONTAP port usage on a storage system

A number of well-known ports are reserved for ONTAP communications with specific services. Port conflicts will occur if a port value in your storage network environment is the same as on ONTAP port.

Network Ports

The following table lists the TCP ports and UDP ports that are used by ONTAP.

<table>
<thead>
<tr>
<th>Service</th>
<th>Port/Protocol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ssh</td>
<td>22/TCP</td>
<td>Secure shell login</td>
</tr>
<tr>
<td>telnet</td>
<td>23/TCP</td>
<td>Remote login</td>
</tr>
<tr>
<td>DNS</td>
<td>53/TCP</td>
<td>Load Balanced DNS</td>
</tr>
<tr>
<td>http</td>
<td>80/TCP</td>
<td>Hyper Text Transfer Protocol</td>
</tr>
<tr>
<td>rpcbind</td>
<td>111/TCP</td>
<td>Remote procedure call</td>
</tr>
<tr>
<td>rpcbind</td>
<td>111/UDP</td>
<td>Remote procedure call</td>
</tr>
<tr>
<td>ntp</td>
<td>123/UDP</td>
<td>Network Time Protocol</td>
</tr>
<tr>
<td>msrpc</td>
<td>135/UDP</td>
<td>MSRPC</td>
</tr>
<tr>
<td>netbios-ssn</td>
<td>139/TCP</td>
<td>NetBIOS service session</td>
</tr>
<tr>
<td>snmp</td>
<td>161/UDP</td>
<td>Simple network management protocol</td>
</tr>
<tr>
<td>https</td>
<td>443/TCP</td>
<td>HTTP over TLS</td>
</tr>
<tr>
<td>microsoft-ds</td>
<td>445/TCP</td>
<td>Microsoft-ds</td>
</tr>
<tr>
<td>mount</td>
<td>635/TCP</td>
<td>NFS mount</td>
</tr>
<tr>
<td>mount</td>
<td>635/UDP</td>
<td>NFS Mount</td>
</tr>
<tr>
<td>named</td>
<td>953/UDP</td>
<td>Name daemon</td>
</tr>
<tr>
<td>nfs</td>
<td>2049/UDP</td>
<td>NFS Server daemon</td>
</tr>
<tr>
<td>nfs</td>
<td>2049/TCP</td>
<td>NFS Server daemon</td>
</tr>
<tr>
<td>nrv</td>
<td>2050/TCP</td>
<td>NetApp Remote Volume protocol</td>
</tr>
<tr>
<td>iscsi</td>
<td>3260/TCP</td>
<td>iSCSI target port</td>
</tr>
<tr>
<td>lockd</td>
<td>4045/TCP</td>
<td>NFS lock daemon</td>
</tr>
<tr>
<td>lockd</td>
<td>4045/UDP</td>
<td>NFS lock daemon</td>
</tr>
<tr>
<td>NFS</td>
<td>4046/TCP</td>
<td>Network Status Monitor</td>
</tr>
<tr>
<td>NSM</td>
<td>4046/UDP</td>
<td>Network Status Monitor</td>
</tr>
<tr>
<td>rquotad</td>
<td>4049/UDP</td>
<td>NFS rquotad protocol</td>
</tr>
<tr>
<td>krb524</td>
<td>4444/UDP</td>
<td>Kerberos 524</td>
</tr>
<tr>
<td>mdns</td>
<td>5353/UDP</td>
<td>Multicast DNS</td>
</tr>
<tr>
<td>Service</td>
<td>Port/Protocol</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>--------------</td>
<td>-------------------------------------------</td>
</tr>
<tr>
<td>HTTPS</td>
<td>5986/UDP</td>
<td>HTTPS Port - Listening binary protocol</td>
</tr>
<tr>
<td>https</td>
<td>8443/TCP</td>
<td>7MTT GUI Tool through https</td>
</tr>
<tr>
<td>ndmp</td>
<td>10000/TCP</td>
<td>Network Data Management Protocol</td>
</tr>
<tr>
<td>Cluster peering</td>
<td>11104/TCP</td>
<td>Cluster peering</td>
</tr>
<tr>
<td>Cluster peering</td>
<td>11105/TCP</td>
<td>Cluster peering</td>
</tr>
<tr>
<td>NDMP</td>
<td>18600 - 18699/TCP</td>
<td>NDMP</td>
</tr>
<tr>
<td>cifs witness port</td>
<td>40001/TCP</td>
<td>cifs witness port</td>
</tr>
<tr>
<td>tls</td>
<td>50000/TCP</td>
<td>Transport layer security</td>
</tr>
<tr>
<td>iscsi</td>
<td>65200/TCP</td>
<td>ISCSI port</td>
</tr>
</tbody>
</table>

**ONTAP Internal Ports**

The following table lists the TCP ports and UDP ports that are used internally by ONTAP. These ports are used to establish intracluster LIF communication:

<table>
<thead>
<tr>
<th>Port/Protocol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>514</td>
<td>Syslog</td>
</tr>
<tr>
<td>900</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>902</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>904</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>905</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>910</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>911</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>913</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>914</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>915</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>918</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>920</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>921</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>924</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>925</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>927</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>928</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>929</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>931</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>Port/Protocol</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------------------</td>
</tr>
<tr>
<td>932</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>933</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>934</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>935</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>936</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>937</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>939</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>940</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>951</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>954</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>955</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>956</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>958</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>961</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>963</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>964</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>966</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>967</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>5125</td>
<td>Alternate Control Port for disk</td>
</tr>
<tr>
<td>5133</td>
<td>Alternate Control Port for disk</td>
</tr>
<tr>
<td>5144</td>
<td>Alternate Control Port for disk</td>
</tr>
<tr>
<td>65502</td>
<td>Node scope SSH</td>
</tr>
<tr>
<td>65503</td>
<td>LIF Sharing</td>
</tr>
<tr>
<td>7810</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>7811</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>7812</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>7813</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>7814</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>7815</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>7816</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>7817</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>7818</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>7819</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>7820</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>7821</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>Port/Protocol</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>7822</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>7823</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>7824</td>
<td>NetApp Cluster RPC</td>
</tr>
<tr>
<td>8023</td>
<td>Node Scope TELNET</td>
</tr>
<tr>
<td>8514</td>
<td>Node Scope RSH</td>
</tr>
<tr>
<td>9877</td>
<td>KMIP Client Port (Internal Local Host Only)</td>
</tr>
</tbody>
</table>
Viewing network information

You can view information related to ports, LIFs, routes, failover rules, failover groups, firewall rules, DNS, NIS, and connections. This information can be useful in situations such as reconfiguring networking settings, or when troubleshooting the cluster.

If you are a cluster administrator, you can view all the available networking information. If you are an SVM administrator, you can view only the information related to your assigned SVMs.

Displaying network port information (cluster administrators only)

You can display information about a specific port, or about all ports on all nodes in the cluster.

About this task

The following information is displayed:

- Node name
- Port name
- IPspace name
- Broadcast domain name
- Link status (up or down)
- MTU setting
- Port speed setting and operational status (1 gigabit or 10 gigabits per second)
- Auto-negotiation setting (true or false)
- Duplex mode and operational status (half or full)
- The port's interface group, if applicable
- The port's VLAN tag information, if applicable
- The port's health status (health or degraded)
- Reasons for a port being marked as degraded

If data for a field is not available (for example, the operational duplex and speed for an inactive port would not be available), the field value is listed as -. 

Step

1. Display network port information by using the network port show command.

You can display detailed information for each port by specifying the -instance parameter, or get specific information by specifying field names using the -fields parameter.

Example

```
cluster-1::> network port show
Node: node1
          Ignore
         Speed(Mbps) Health  Health
```
Displaying information about a VLAN (cluster administrators only)

You can display information about a specific VLAN or about all VLANs in the cluster.

About this task

You can display detailed information for each VLAN by specifying the `-instance` parameter. You can display specific information by specifying field names using the `-fields` parameter.

Step

1. Display information about VLANs by using the `network port vlan show` command.

Example

The following command displays information about all VLANs in the cluster:

```
cluster-1::> network port vlan show
```

<table>
<thead>
<tr>
<th>Node</th>
<th>VLAN Name</th>
<th>Port</th>
<th>VLAN ID</th>
<th>MAC Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>cluster-1-01</td>
<td>a0a-10</td>
<td>a0a</td>
<td>10</td>
<td>02:a0:98:06:10:b2</td>
</tr>
<tr>
<td></td>
<td>a0a-20</td>
<td>a0a</td>
<td>20</td>
<td>02:a0:98:06:10:b2</td>
</tr>
<tr>
<td></td>
<td>a0a-30</td>
<td>a0a</td>
<td>30</td>
<td>02:a0:98:06:10:b2</td>
</tr>
<tr>
<td></td>
<td>a0a-40</td>
<td>a0a</td>
<td>40</td>
<td>02:a0:98:06:10:b2</td>
</tr>
<tr>
<td></td>
<td>a0a-50</td>
<td>a0a</td>
<td>50</td>
<td>02:a0:98:06:10:b2</td>
</tr>
<tr>
<td>cluster-1-02</td>
<td>a0a-10</td>
<td>a0a</td>
<td>10</td>
<td>02:a0:98:06:10:ca</td>
</tr>
<tr>
<td></td>
<td>a0a-20</td>
<td>a0a</td>
<td>20</td>
<td>02:a0:98:06:10:ca</td>
</tr>
<tr>
<td></td>
<td>a0a-30</td>
<td>a0a</td>
<td>30</td>
<td>02:a0:98:06:10:ca</td>
</tr>
<tr>
<td></td>
<td>a0a-40</td>
<td>a0a</td>
<td>40</td>
<td>02:a0:98:06:10:ca</td>
</tr>
<tr>
<td></td>
<td>a0a-50</td>
<td>a0a</td>
<td>50</td>
<td>02:a0:98:06:10:ca</td>
</tr>
</tbody>
</table>

Related tasks

`Creating a VLAN` on page 35
Displaying interface group information (cluster administrators only)

You can display information about an interface group to determine its configuration.

About this task

The following information is displayed:

- Node on which the interface group is located
- List of network ports that are included in the interface group
- Interface group's name
- Distribution function (MAC, IP, port, or sequential)
- Interface group's Media Access Control (MAC) address
- Port activity status; that is, whether all aggregated ports are active (full participation), whether some are active (partial participation), or whether none are active

Step

1. Display information about interface groups by using the `network port ifgrp show` command.

You can display detailed information for each node by specifying the `-instance` parameter. You can display specific information by specifying field names using the `-fields` parameter.

Example

The following command displays information about all interface groups in the cluster:

```
cluster-1::> network port ifgrp show
```

<table>
<thead>
<tr>
<th>Node</th>
<th>IfGrp</th>
<th>Distribution Function</th>
<th>MAC Address</th>
<th>Active Ports</th>
<th>Ports</th>
</tr>
</thead>
<tbody>
<tr>
<td>cluster-1-01</td>
<td>a0a</td>
<td>ip</td>
<td>02:a0:98:06:10:b2</td>
<td>full</td>
<td>e7a, e7b</td>
</tr>
<tr>
<td>cluster-1-02</td>
<td>a0a</td>
<td>sequential</td>
<td>02:a0:98:06:10:ca</td>
<td>full</td>
<td>e7a, e7b</td>
</tr>
<tr>
<td>cluster-1-03</td>
<td>a0a</td>
<td>port</td>
<td>02:a0:98:08:5b:66</td>
<td>full</td>
<td>e7a, e7b</td>
</tr>
<tr>
<td>cluster-1-04</td>
<td>a0a</td>
<td>mac</td>
<td>02:a0:98:08:61:4e</td>
<td>full</td>
<td>e7a, e7b</td>
</tr>
</tbody>
</table>

The following command displays detailed interface group information for a single node:

```
cluster-1::> network port ifgrp show -instance -node cluster-1-01
```

Node: cluster-1-01
Interface Group Name: a0a
Distribution Function: ip
Create Policy: multimode
MAC Address: 02:a0:98:06:10:b2
Port Participation: full
Network Ports: e7a, e7b
Up Ports: e7a, e7b
Down Ports: -
Displaying LIF information

You can view detailed information about a LIF to determine its configuration. You might also want to view this information to diagnose basic LIF problems, such as checking for duplicate IP addresses or verifying whether the network port belongs to the correct subnet. storage virtual machine (SVM) administrators can view only the information about the LIFs associated with the SVM.

About this task

The following information is displayed:

- IP address associated with the LIF
- Administrative status of the LIF
- Operational status of the LIF
  - The operational status of data LIFs is determined by the status of the SVM with which the data LIFs are associated. When the SVM is stopped, the operational status of the LIF changes to down. When the SVM is started again, the operational status changes to up.
- Node and the port on which the LIF resides

If data for a field is not available (for example, if there is no extended status information), the field value is listed as -.

Step

1. Display LIF information by using the network interface show command.

   You can view detailed information for each LIF by specifying the -instance parameter, or get specific information by specifying field names using the -fields parameter.

Example

The following command displays general information about all LIFs in a cluster:

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

Logical     Status    Network          Current     Current Is
Vserver     Interface  Admin/Oper Address/Mask     Node         Port    Home
---------- ---------- ---------- ---------------- ------------ ------- ----
example     lif1       up/up    192.0.2.129/22   node-01     e0d     false
node-01     cluster_mgmt up/up 192.0.2.3/20     node-02     e0c     false
            clus1      up/up    192.0.2.65/18   node-01     e0a     true
            clus2      up/up    192.0.2.66/18   node-01     e0b     true
            mgmt1      up/up    192.0.2.1/20    node-01     e0c     true
node-02     clus1      up/up    192.0.2.67/18   node-02     e0a     true
            clus2      up/up    192.0.2.68/18   node-02     e0b     true
            mgmt2      up/up    192.0.2.2/20    node-02     e0d     true
```

Related tasks

- Creating an interface group on page 32
- Adding a port to an interface group on page 33
- Deleting an interface group on page 34
The following command shows detailed information about a single LIF:

```
vs1:/> network interface show -lif data1 -instance
```

```
Vserver Name: vs1
Logical Interface Name: data1
Role: data
Data Protocol: nfs,cifs
Home Node: node-01
Home Port: e0c
Current Node: node-03
Current Port: e0c
Operational Status: up
Extended Status: -
Is Home: false
Network Address: 192.0.2.128
Netmask: 255.255.192.0
Bits in the Netmask: 18
IPv4 Link Local: -
Subnet Name: -
Administrative Status: up
Failover Policy: local-only
Firewall Policy: data
Auto Revert: false
Fully Qualified DNS Zone Name: xxx.example.com
DNS Query Listen Enable: false
Failover Group Name: Default
FCP WWPN: -
Address family: ipv4
Comment: -
IPspace of LIF: Default
```

Related tasks

- Creating a LIF on page 72
- Modifying a LIF on page 75
- Migrating a LIF on page 76
- Reverting a LIF to its home port on page 78
- Deleting a LIF on page 78

Displaying routing information

You can display information about all routing groups and static routes within an SVM.

**Step**

1. Depending on the type of routing information that you want to view, enter the applicable command:

<table>
<thead>
<tr>
<th>To view information about...</th>
<th>Enter...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static routes, per SVM</td>
<td><code>network route show</code></td>
</tr>
<tr>
<td>LIFs on each route, per SVM</td>
<td><code>network route show-lifs</code></td>
</tr>
</tbody>
</table>
You can display detailed information for each static route or routing group by specifying the -instance parameter.

**Example**

The following command displays the static routes within the SVMs in cluster-1:

```bash
cluster-1::> network route show
Vserver             Destination     Gateway         Metric
------------------- --------------- --------------- ------
Cluster             0.0.0.0/0       10.63.0.1       10
cluster-1           0.0.0.0/0       198.51.9.1      10
vs1                 0.0.0.0/0       192.0.2.1       20
vs3                 0.0.0.0/0       192.0.2.1       20
```

The following command displays the association of static routes and logical interfaces (LIFs) within all SVMs in cluster-1:

```bash
cluster-1::> network route show-lifs
Vserver: Cluster
Destination             Gateway                 Logical Interfaces
----------------------  ----------------------  -------------------
0.0.0.0/0               10.63.0.1               -
Vserver: cluster-1
Destination             Gateway                 Logical Interfaces
----------------------  ----------------------  -------------------
0.0.0.0/0               198.51.9.1              cluster_mgmt,
cluster-1_mgmt1,
Vserver: vs1
Destination             Gateway                 Logical Interfaces
----------------------  ----------------------  -------------------
0.0.0.0/0               192.0.2.1               data1_1, data1_2
Vserver: vs3
Destination             Gateway                 Logical Interfaces
----------------------  ----------------------  -------------------
0.0.0.0/0               192.0.2.1               data2_1, data2_2
```

**Related concepts**

*Managing routing in an SVM* on page 101

**Related tasks**

*Creating a static route* on page 101
Displaying DNS host table entries (cluster administrators only)

The DNS host table entries map host names to IP addresses. You can display the host names and alias names and the IP address that they map to for all SVMs in a cluster.

**Step**

1. Display the host name entries for all SVMs by using the `vserver services name-service dns hosts show` command.

**Example**

The following example displays the host table entries:

```
cluster-1::> vserver services name-service dns hosts show

<table>
<thead>
<tr>
<th>Vserver</th>
<th>Address</th>
<th>Hostname</th>
<th>Aliases</th>
</tr>
</thead>
<tbody>
<tr>
<td>cluster-1</td>
<td>10.72.219.36</td>
<td>lnx219-36</td>
<td>-</td>
</tr>
<tr>
<td>vs1</td>
<td>10.72.219.37</td>
<td>lnx219-37</td>
<td>lnx219-37.example.com</td>
</tr>
</tbody>
</table>
```

**Related tasks**

*Configuring an SVM and data LIFs for host-name resolution using an external DNS server* on page 83

Displaying DNS domain configurations

You can display the DNS domain configuration of one or more storage virtual machines (SVMs) in your cluster to verify that it is configured properly.

**Step**

1. Viewing 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

<table>
<thead>
<tr>
<th>Vserver</th>
<th>State</th>
<th>Domains</th>
<th>Name Servers</th>
</tr>
</thead>
<tbody>
<tr>
<td>cluster-1</td>
<td>enabled</td>
<td>xyz.company.com</td>
<td>192.56.0.129, 192.56.0.130</td>
</tr>
<tr>
<td>vs1</td>
<td>enabled</td>
<td>xyz.company.com</td>
<td>192.56.0.129, 192.56.0.130</td>
</tr>
<tr>
<td>vs2</td>
<td>enabled</td>
<td>xyz.company.com</td>
<td>192.56.0.129, 192.56.0.130</td>
</tr>
<tr>
<td>vs3</td>
<td>enabled</td>
<td>xyz.company.com</td>
<td>192.56.0.129, 192.56.0.130</td>
</tr>
</tbody>
</table>
```

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

```
```
Displaying information about failover groups

You can view information about failover groups, including the list of nodes and ports in each failover group, whether failover is enabled or disabled, and the type of failover policy that is being applied to each LIF.

Steps

1. Display the target ports for each failover group by using the `network interface failover-groups show` command.

   **Example**
   
The following command displays information about all failover groups on a two-node cluster:

   ```
   cluster-1::> network interface failover-groups show
   Vserver Group Targets
   ----------------- ----------------------------------
   Cluster Cluster
   vs1 Default
   ```

2. Display the target ports and broadcast domain for a specific failover group by using the `network interface failover-groups show` command.

   **Example**
   
The following command displays detailed information about failover group data12 for SVM vs4:

   ```
   cluster-1::> network interface failover-groups show -vserver vs4 -failover-group data12
   Vserver Name: vs4
   Failover Group Name: data12
   Failover Targets: cluster1-01:e0f, cluster1-02:e0f,
   cluster1-01:e0g, cluster1-02:e0g
   Broadcast Domain: Default
   ```

3. Display the failover settings used by all LIFs by using the `network interface show` command.

   **Example**
   
The following command displays the failover policy and failover group that is being used by each LIF:
Related tasks

Creating a failover group on page 56
Configuring failover settings on a LIF on page 57

Related references

Commands for managing failover groups and policies on page 59

Displaying LIF failover targets

You might have to check whether the failover policies and the failover groups of a LIF are configured correctly. To prevent misconfiguration of the failover rules, you can display the failover targets for a single LIF or for all LIFs.

About this task

Displaying LIF failover targets enables you to check for the following:

- Whether the LIFs are configured with the correct failover group and failover policy
- Whether the resulting list of failover target ports is appropriate for each LIF
- Whether the failover target of a data LIF is not a management port (e0M)

Step

1. Display the failover targets of a LIF by using the failover option of the network interface show command.

Example

The following command displays information about the failover targets for all LIFs in a two-node cluster. The Failover Targets row shows the (prioritized) list of node-port combinations for a given LIF.
Related tasks

*Creating a failover group* on page 56

*Configuring failover settings on a LIF* on page 57

Related references

*Commands for managing failover groups and policies* on page 59

### Displaying LIFs in a load balancing zone

You can verify whether a load balancing zone is configured correctly by displaying all of the LIFs that belong to it. You can also view the load balancing zone of a particular LIF, or the load balancing zones for all LIFs.

**Step**

1. Display the LIFs and load balancing details that you want by using one of the following commands:

   **To display...** | **Enter...**
   --- | ---
   LIFs in a particular load balancing zone | `network interface show -dns-zone zone_name`
   The load balancing zone of a particular LIF | `network interface show -lif lif_name -fields dns-zone`
   The load balancing zones of all LIFs | `network interface show -fields dns-zone`

**Examples of displaying load balancing zones for LIFs**

The following command displays the details of all LIFs in the load balancing zone storage.company.com for SVM vs0:

```bash
cluster-1::> net int show -vserver vs0 -dns-zone storage.company.com
```

<table>
<thead>
<tr>
<th>Logical Vserver</th>
<th>Interface</th>
<th>Status</th>
<th>Network</th>
<th>Current Node</th>
<th>Current Is Port</th>
<th>Home</th>
</tr>
</thead>
<tbody>
<tr>
<td>vs0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The following command displays the DNS zone details of the LIF data3:

```
cluster-1::> network interface show -lif data3 -fields dns-zone
Vserver  lif    dns-zone
-------  -----  --------
vs0      data3  storage.company.com
```

The following command displays the list of all LIFs in the cluster and their corresponding DNS zones:

```
cluster-1::> network interface show -fields dns-zone
Vserver   lif          dns-zone
-------   ------------ --------
cluster   cluster_mgmt none
ndeux-21  clus1        none
ndeux-21  clus2        none
ndeux-21  mgmt1        none
vs0       data1        storage.company.com
vs0       data2        storage.company.com
```

Related tasks

Displaying LIF information on page 121

Displaying cluster connections

You can display all the active connections in the cluster or a count of active connections on the node by client, logical interface, protocol, or service. You can also display all the listening connections in the cluster.

Displaying active connections by client (cluster administrators only)

You can view the active connections by client to verify the node that a specific client is using and to view possible imbalances between client counts per node.

About this task

The count of active connections by client is useful in the following scenarios:

- Finding a busy or overloaded node.
- Determining why a particular client’s access to a volume is slow.
  You can view details about the node that the client is accessing and then compare it with the node on which the volume resides. If accessing the volume requires traversing the cluster network, clients might experience decreased performance because of the remote access to the volume on an oversubscribed remote node.
- Verifying that all nodes are being used equally for data access.
- Finding clients that have an unexpectedly high number of connections.
- Verifying whether certain clients have connections to a node.
Step

1. Display a count of the active connections by client on a node by using the `network connections active show-clients` command.

   For more information about this command, see the man page.

   **Example**

   ```
   cluster-1::> network connections active show-clients
   Node  Vserver Name    Client IP Address     Count
   ------  --------------  -----------------     ------
   node0   vs0             192.0.2.253                1
   vs0             192.0.2.252                2
   Cluster         192.10.2.124               5
   node1   vs0             192.0.2.250                1
   vs0             192.0.2.252                3
   Cluster         192.10.2.123               4
   node2   vs1             customer.example.com       1
   vs1             192.0.2.245                3
   Cluster         192.10.2.122               4
   node3   vs1             customer.example.org       1
   vs1             customer.example.net       3
   Cluster         192.10.2.121               4
   ```

2. Displaying active connections by protocol (cluster administrators only)

   You can display a count of the active connections by protocol (TCP or UDP) on a node to compare the usage of protocols within the cluster.

   **About this task**

   The count of active connections by protocol is useful in the following scenarios:

   - Finding the UDP clients that are losing their connection.
     If a node is near its connection limit, UDP clients are the first to be dropped.
   - Verifying that no other protocols are being used.

   **Step**

   1. Display a count of the active connections by protocol on a node by using the `network connections active show-protocols` command.

      For more information about this command, see the man page.

   **Example**

   ```
   cluster-1::> network connections active show-protocols
   Node     Vserver Name  Protocol   Count
   -------  ------------  ---------  ------
   node0   vs0           UDP            19
          Cluster       TCP            11
   node1   vs0           UDP            17
          Cluster       TCP             8
   node2   vs1           UDP            14
          Cluster       TCP             8
   node3   vs1           UDP            18
          Cluster       TCP             4
   ```
Displaying active connections by service (cluster administrators only)

You can display a count of the active connections by service type (for example, by NFS, CIFS, mount, and so on) for each node in a cluster. This is useful to compare the usage of services within the cluster, which helps to determine the primary workload of a node.

About this task

The count of active connections by service is useful in the following scenarios:

- Verifying that all nodes are being used for the appropriate services and that the load balancing for that service is working.
- Verifying that no other services are being used.

Step

1. Display a count of the active connections by service on a node by using the `network connections active show-services` command.

   For more information about this command, see the man page.

Example

```
cluster-1::> network connections active show-services

<table>
<thead>
<tr>
<th>Node</th>
<th>Vserver Name</th>
<th>Service</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>node0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>vs0</td>
<td>mount</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>vs0</td>
<td>nfs</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>vs0</td>
<td>nlm_v4</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>vs0</td>
<td>cifs_srv</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>vs0</td>
<td>port_map</td>
<td>18</td>
</tr>
<tr>
<td></td>
<td>vs0</td>
<td>rclopcp</td>
<td>27</td>
</tr>
<tr>
<td></td>
<td>Cluster</td>
<td>ctlopcp</td>
<td>60</td>
</tr>
<tr>
<td>node1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>vs0</td>
<td>cifs_srv</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>vs0</td>
<td>rclopcp</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>Cluster</td>
<td>ctlopcp</td>
<td>60</td>
</tr>
<tr>
<td>node2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>vs1</td>
<td>rclopcp</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>Cluster</td>
<td>ctlopcp</td>
<td>60</td>
</tr>
<tr>
<td>node3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>vs1</td>
<td>cifs_srv</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>vs1</td>
<td>rclopcp</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>Cluster</td>
<td>ctlopcp</td>
<td>60</td>
</tr>
</tbody>
</table>
```

Displaying active connections by LIF on a node and SVM

You can display a count of active connections for each LIF, by node and storage virtual machine (SVM), to view connection imbalances between LIFs within the cluster.

About this task

The count of active connections by LIF is useful in the following scenarios:

- Finding an overloaded LIF by comparing the number of connections on each LIF.
- Verifying that DNS load balancing is working for all data LIFs.
- Comparing the number of connections to the various SVMs to find the SVMs that are used the most.
Step

1. Display a count of active connections for each LIF by SVM and node by using the `network connections active show-lifs` command.

For more information about this command, see the man page.

Example

```
cluster-1::> network connections active show-lifs
Node Vserver Name Interface Name Count
-------- --------------- ----------- ----
node0
  vs0  datalif1 3
  Cluster node0_clus_1 6
  Cluster node0_clus_2 5
node1
  vs0  datalif2 3
  Cluster node1_clus_1 3
  Cluster node1_clus_2 5
node2
  vs1  datalif2 1
  Cluster node2_clus_1 5
  Cluster node2_clus_2 3
node3
  vs1  datalif1 1
  Cluster node3_clus_1 2
  Cluster node3_clus_2 2
```

Displaying active connections in a cluster

You can display information about the active connections in a cluster to view the LIF, port, remote host, service, storage virtual machines (SVMs), and protocol used by individual connections.

About this task

Viewing the active connections in a cluster is useful in the following scenarios:

- Verifying that individual clients are using the correct protocol and service on the correct node.
- If a client is having trouble accessing data using a certain combination of node, protocol, and service, you can use this command to find a similar client for configuration or packet trace comparison.

Step

1. Display the active connections in a cluster by using the `network connections active show` command.

For more information about this command, see the man page.

Example

The following command shows the active connections on the node node1:

```
cluster-1::> network connections active show -node node1
Vserver Interface Name:Local Port Remote Host:Port Protocol/Service
-------- ------------------ ----------- ------------ ---------------
Node: node1
  Cluster node1_clus_1:50297 192.0.2.253:7700 TCP/ctlopcp
  Cluster node1_clus_1:13387 192.0.2.253:7700 TCP/ctlopcp
  Cluster node1_clus_1:8340 192.0.2.253:7700 TCP/ctlopcp
  Cluster node1_clus_1:42766 192.0.2.253:7700 TCP/ctlopcp
  Cluster node1_clus_1:36119 192.0.2.253:7700 TCP/ctlopcp
```
The following command shows the active connections on SVM vs1:

```
cluster-1::> network connections active show -vserver vs1
```

<table>
<thead>
<tr>
<th>Vserver Name</th>
<th>Interface Name:Local Port</th>
<th>Remote Host:Port</th>
<th>Protocol/Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>vs1</td>
<td>data1:111</td>
<td>host1.aa.com:10741</td>
<td>UDP/port-map</td>
</tr>
<tr>
<td>vs1</td>
<td>data1:111</td>
<td>host1.aa.com:12017</td>
<td>UDP/port-map</td>
</tr>
</tbody>
</table>

### Displaying listening connections in a cluster

You can display information about the listening connections in a cluster to view the LIFs and ports that are accepting connections for a given protocol and service.

**About this task**

Viewing the listening connections in a cluster is useful in the following scenarios:

- Verifying that the desired protocol or service is listening on a LIF if client connections to that LIF fail consistently.
- Verifying that a UDP/rclopcp listener is opened at each cluster LIF if remote data access to a volume on one node through a LIF on another node fails.
- Verifying that a UDP/rclopcp listener is opened at each cluster LIF if SnapMirror transfers between two nodes in the same cluster fail.
- Verifying that a TCP/ctlopcp listener is opened at each intercluster LIF if SnapMirror transfers between two nodes in different clusters fail.

**Step**

1. Display the listening connections per node by using the `network connections listening show` command.

**Example**

```
cluster-1::> network connections listening show
```

<table>
<thead>
<tr>
<th>Vserver Name</th>
<th>Interface Name:Local Port</th>
<th>Protocol/Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cluster</td>
<td>node0_clus_1:7700</td>
<td>TCP/ctlopcp</td>
</tr>
<tr>
<td>vs1</td>
<td>data1:4049</td>
<td>UDP/unknown</td>
</tr>
<tr>
<td>vs1</td>
<td>data1:111</td>
<td>TCP/port-map</td>
</tr>
<tr>
<td>vs1</td>
<td>data1:111</td>
<td>UDP/port-map</td>
</tr>
<tr>
<td>vs1</td>
<td>data1:4046</td>
<td>TCP/sm</td>
</tr>
<tr>
<td>vs1</td>
<td>data1:4046</td>
<td>UDP/sm</td>
</tr>
<tr>
<td>vs1</td>
<td>data1:4045</td>
<td>TCP/nlm-v4</td>
</tr>
<tr>
<td>vs1</td>
<td>data1:4045</td>
<td>UDP/nlm-v4</td>
</tr>
<tr>
<td>vs1</td>
<td>data1:2049</td>
<td>TCP/nfs</td>
</tr>
<tr>
<td>vs1</td>
<td>data1:2049</td>
<td>UDP/nfs</td>
</tr>
<tr>
<td>vs1</td>
<td>data1:635</td>
<td>TCP/mount</td>
</tr>
<tr>
<td>vs1</td>
<td>data1:635</td>
<td>UDP/mount</td>
</tr>
<tr>
<td>Cluster</td>
<td>node0_clus_2:7700</td>
<td>TCP/ctlopcp</td>
</tr>
</tbody>
</table>
## Commands for diagnosing network problems

You can diagnose problems on your network by using commands such as `ping`, `traceroute`, `ndp`, and `tcpdump`. You can also use commands such as `ping6` and `traceroute6` to diagnose IPv6 problems.

<table>
<thead>
<tr>
<th>If you want to...</th>
<th>Enter this command...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test whether the node can reach other hosts on</td>
<td><code>network ping</code></td>
</tr>
<tr>
<td>your network</td>
<td></td>
</tr>
<tr>
<td>Test whether the node can reach other hosts on</td>
<td><code>network ping6</code></td>
</tr>
<tr>
<td>your IPv6 network</td>
<td></td>
</tr>
<tr>
<td>Trace the route that the IPv4 packets take to</td>
<td><code>network traceroute</code></td>
</tr>
<tr>
<td>a network node</td>
<td></td>
</tr>
<tr>
<td>Trace the route that the IPv6 packets take to</td>
<td><code>network traceroute6</code></td>
</tr>
<tr>
<td>a network node</td>
<td></td>
</tr>
<tr>
<td>Manage the Neighbor Discovery Protocol</td>
<td><code>network ndp</code></td>
</tr>
<tr>
<td>(NDP)</td>
<td></td>
</tr>
<tr>
<td>Display statistics about packets that are</td>
<td><code>run -node node_name ifstat</code></td>
</tr>
<tr>
<td>received and sent on a specified network</td>
<td></td>
</tr>
<tr>
<td>interface or on all network interfaces</td>
<td><strong>Note:</strong> This command is available</td>
</tr>
<tr>
<td></td>
<td>from the nodeshell.</td>
</tr>
<tr>
<td>Display information about neighboring devices</td>
<td><code>network device-discovery show</code></td>
</tr>
<tr>
<td>that are discovered from each node and port in</td>
<td></td>
</tr>
<tr>
<td>the cluster, including the remote device type</td>
<td></td>
</tr>
<tr>
<td>and device platform</td>
<td></td>
</tr>
<tr>
<td>View the CDP neighbors of the node (ONTAP</td>
<td>`run -node node_name cdpd show-</td>
</tr>
<tr>
<td>supports only CDPv1 advertisements)</td>
<td>neighbors`</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> This command is available</td>
</tr>
<tr>
<td></td>
<td>from the nodeshell.</td>
</tr>
<tr>
<td>Trace the packets that are sent and received</td>
<td>`network tcpdump start -node node-</td>
</tr>
<tr>
<td>in the network</td>
<td>name -port port_name`</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> This command is available</td>
</tr>
<tr>
<td></td>
<td>from the nodeshell.</td>
</tr>
<tr>
<td>Measure latency and throughput between inter-</td>
<td>`network test-path -source-node</td>
</tr>
<tr>
<td>cluster or intracluster nodes</td>
<td><code>source_nodename</code></td>
</tr>
<tr>
<td></td>
<td><code>cluster destination_clustername</code> -</td>
</tr>
<tr>
<td></td>
<td><code>destination-node</code></td>
</tr>
<tr>
<td></td>
<td><code>destination_nodename</code> -session-type</td>
</tr>
<tr>
<td></td>
<td>`Default</td>
</tr>
<tr>
<td></td>
<td>`AsyncMirrorRemote</td>
</tr>
<tr>
<td></td>
<td><code>RemoteDataTransfer</code></td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> For more information, see the</td>
</tr>
<tr>
<td></td>
<td><em>Performance Monitoring Power Guide.</em></td>
</tr>
</tbody>
</table>
Displaying network connectivity with neighbor discovery protocols

In a data center, you can use neighbor discovery protocols to view network connectivity between a pair of physical or virtual systems and their network interfaces. ONTAP supports two neighbor discovery protocols: Cisco Discovery Protocol (CDP) and, starting in ONTAP 9, Link Layer Discovery Protocol (LLDP).

About this task
Neighbor discovery protocols enable you to automatically discover and view information about directly connected protocol-enabled devices in a network. Each device advertises identification, capabilities, and connectivity information. This information is transmitted in Ethernet frames to a multicast MAC address and is received by all neighboring protocol-enabled devices.

For two devices to become neighbors, each must have a protocol enabled and correctly configured. Discovery protocol functionality is limited to directly connected networks. Neighbors can include protocol-enabled devices such as switches, routers, bridges, and so on. ONTAP supports two neighbor discovery protocols, which can be used individually or together.

Cisco Discovery Protocol (CDP)
CDP is a proprietary link layer protocol developed by Cisco Systems. It is enabled by default in ONTAP for cluster ports, but must be enabled explicitly for data ports.

Link Layer Discovery Protocol (LLDP)
LLDP is a vendor-neutral protocol specified in the standards document IEEE 802.1AB. It must be enabled explicitly for all ports.

Using CDP to detect network connectivity
Using CDP to detect network connectivity consists of reviewing deployment considerations, enabling it on data ports, viewing neighbor devices, and adjusting CDP configuration values as needed. CDP is enabled by default on cluster ports.

Before you begin
CDP must also be enabled on any switches and routers before information about neighbor devices can be displayed.

About this task
CDP is also used by the cluster switch health monitor to automatically discover your cluster and management network switches.

Considerations for using CDP
By default, CDP-compliant devices send CDPv2 advertisements. CDP-compliant devices send CDPv1 advertisements only when they receive CDPv1 advertisements. ONTAP supports only
CDPv1. Therefore, when an ONTAP node sends CDPv1 advertisements, CDP-compliant neighboring devices send back CDPv1 advertisements.

You should consider the following information before enabling CDP on a node:

- CDP is always enabled on cluster ports.
- CDP is disabled, by default, on all non-cluster ports.
- CDP is supported for all ports.
- CDP advertisements are sent and received by ports that are in the up state.
- CDP must be enabled on both the transmitting and receiving devices for sending and receiving CDP advertisements.
- CDP advertisements are sent at regular intervals, and you can configure the time interval.
- When IP addresses are changed for a LIF, the node sends the updated information in the next CDP advertisement.
  
  Note: Sometimes when LIFs are changed on the node, the CDP information is not updated at the receiving device side (for example, a switch). If you encounter such a problem, you should configure the network interface of the node to the down status and then to the up status.

- Only IPv4 addresses are advertised in CDP advertisements.
- For physical network ports with VLANs, all of the LIFs configured on the VLANs on that port are advertised.
- For physical ports that are part of an interface group, all of the IP addresses configured on that interface group are advertised on each physical port.
- For an interface group that hosts VLANs, all of the LIFs configured on the interface group and the VLANs are advertised on each of the network ports.
- For packets with MTU size equal to or greater than 1,500 bytes, only the number of LIFs that can fit into a 1500 MTU-sized packet is advertised.
- Some Cisco switches always send CDP packets that are tagged on VLAN 1 if the native (default) VLAN of a trunk is anything other than 1. ONTAP only supports CDP packets that are untagged, both for sending and receiving. This results in storage platforms running ONTAP being visible to Cisco devices (using the show cdp neighbors command), and only the Cisco devices that send untagged CDP packets are visible to ONTAP.

### Enabling or disabling CDP

To discover and send advertisements to CDP-compliant neighboring devices, CDP must be enabled on each node of the cluster. By default, CDP is enabled on all cluster ports of a node and disabled on all non-cluster ports of a node.

#### About this task

The `cdpd.enable` option controls whether CDP is enabled or disabled on the ports of a node:

- **on** enables CDP on non-cluster ports.
- **off** disables CDP on non-cluster ports; you cannot disable CDP on cluster ports.

When CDP is disabled on a port that is connected to a CDP-compliant device, network traffic might not be optimized.
Steps

1. Display the current CDP setting for a node, or for all nodes in a cluster:

   To view the CDP setting of... Enter...
   
   A node run -node node_name options cdpd.enable
   
   All nodes in a cluster options cdpd.enable

2. Enable or disable CDP on all ports of a node, or on all ports of all nodes in a cluster:

   To enable or disable CDP on...
   
   A node run -node node_name options cdpd.enable {on|off}
   
   All nodes in a cluster options cdpd.enable {on|off}

Viewing CDP neighbor information

You can view information about the neighboring devices that are connected to each port of the nodes of your cluster, provided that the port is connected to a CDP-compliant device. You can use the `cdpd show-neighbors` command to view neighbor information. The `network device-discovery show` command can also be used to display information about neighboring devices.

About this task

Because CDP is always enabled for cluster ports, CDP neighbor information is always displayed for those ports. CDP must be enabled on non-cluster ports for neighbor information to appear for those ports.

Some Cisco switches always send CDP packets that are tagged on VLAN 1 if the native (default) VLAN of a trunk is anything other than 1.

ONTAP supports only CDP packets that are untagged, for sending and receiving. This means that nodes that are running ONTAP are visible to Cisco devices (using the `show cdp neighbors` command), but only the Cisco devices that send untagged CDP packets are visible to ONTAP.

Step

1. Display information about all CDP-compliant devices that are connected to the ports on a node in the cluster:

   run -node node_name cdpd show-neighbors

Example

The following command shows the neighbors that are connected to the ports on node cluster-1_01:

```
cluster-1::> run -node cluster-1_01 cdpd show-neighbors
Local  Remote          Remote               Remote          Hold  Remote
Port  Device          Interface            Platform        Time  Capability
----- --------------- -------------------- --------------- ----- ----------
e0a   sw-215-cr(4C2)  GigabitEthernet1/17  cisco WS-C4948   125   RSI

e0b   sw-215-11(4C5)  GigabitEthernet1/15  cisco WS-C4948   145   SI

e0c   sw-215-11(4C5)  GigabitEthernet1/16  cisco WS-C4948   145   SI
```

The output lists the Cisco devices that are connected to each port of the specified node. The Remote Capability column specifies the capabilities of each remote device. The following capabilities are available:
• R—Router
• T—Transparent bridge
• B—Source-route bridge
• S—Switch
• H—Host
• I—IGMP
• r—Repeater
• P—Phone

Configuring the hold time for CDP messages

Hold time is the period of time for which CDP advertisements are stored in cache in neighboring CDP-compliant devices. Hold time is advertised in each CDPv1 packet and is updated whenever a CDPv1 packet is received by a node.

About this task

• The value of the `cdpd.holdtime` option should be set to the same value on both nodes of an HA pair.
• The default hold time value is 180 seconds, but you can enter values ranging from 10 seconds to 255 seconds.
• If an IP address is removed before the hold time expires, the CDP information is cached until the hold time expires.

Steps

1. Display the current CDP hold time for a node, or for all nodes in a cluster:

<table>
<thead>
<tr>
<th>To view the hold time of...</th>
<th>Enter...</th>
</tr>
</thead>
<tbody>
<tr>
<td>A node</td>
<td><code>run -node node_name options cdpd.holdtime</code></td>
</tr>
<tr>
<td>All nodes in a cluster</td>
<td><code>options cdpd.holdtime</code></td>
</tr>
</tbody>
</table>

2. Configure the CDP hold time on all ports of a node, or on all ports of all nodes in a cluster:

<table>
<thead>
<tr>
<th>To set the hold time on...</th>
<th>Enter...</th>
</tr>
</thead>
<tbody>
<tr>
<td>A node</td>
<td><code>run -node node_name options cdpd.holdtime holdtime</code></td>
</tr>
<tr>
<td>All nodes in a cluster</td>
<td><code>options cdpd.holdtime holdtime</code></td>
</tr>
</tbody>
</table>

Setting the interval for sending CDP advertisements

CDP advertisements are sent to CDP neighbors at periodic intervals. You can increase or decrease the interval for sending CDP advertisements depending on network traffic and changes in the network topology.

About this task

• The value of the `cdpd.interval` option should be set to the same value on both nodes of an HA pair.
• The default interval is 60 seconds, but you can enter a value from 5 seconds to 900 seconds.

Steps
1. Display the current CDP advertisement time interval for a node, or for all nodes in a cluster:

   To view the interval for... Enter...
   A node run -node node_name options cdpd.interval
   All nodes in a cluster options cdpd.interval

2. Configure the interval for sending CDP advertisements for all ports of a node, or for all ports of all nodes in a cluster:

   To set the interval for... Enter...
   A node run -node node_name options cdpd.interval interval
   All nodes in a cluster options cdpd.interval interval

Viewing or clearing CDP statistics

You can view the CDP statistics for the cluster and non-cluster ports on each node to detect potential network connectivity issues. CDP statistics are cumulative from the time they were last cleared.

About this task

Because CDP is always enabled for cluster ports, CDP statistics are always displayed for traffic on those ports. CDP must be enabled on non-cluster ports for statistics to appear for those ports.

Step
1. Display or clear the current CDP statistics for all ports on a node:

   If you want to... Enter...
   View the CDP statistics run -node node_name cdpd show-stats
   Clear the CDP statistics run -node node_name cdpd zero-stats

Example of showing and clearing statistics

The following command shows the CDP statistics before they are cleared. The output displays the total number of packets that have been sent and received since the last time the statistics were cleared.

```
cluster-1::> run -node node1 cdpd show-stats
RECEIVE
Packets: 9116 | Csum Errors: 0 | Unsupported Vers: 4561
Invalid length: 0 | Malformed: 0 | Mem alloc fails: 0
Missing TLVs: 0 | Cache overflow: 0 | Other errors: 0

TRANSMIT
Packets: 4557 | Xmit fails: 0 | No hostname: 0
Packet truncated: 0 | Mem alloc fails: 0 | Other errors: 0

OTHER
Init failures: 0
```

The following command clears the CDP statistics:

```
cluster-1::> run -node node1 cdpd zero-stats
```
The following command shows the statistics after they are cleared:

```
cluster-1::> run -node node1 cdpd show-stats
```

<table>
<thead>
<tr>
<th>RECEIVE</th>
<th></th>
<th>Csum Errors:</th>
<th></th>
<th>Unsupported Vers:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Packets:</td>
<td>0</td>
<td>Malformed:</td>
<td>0</td>
<td>Mem alloc fails:</td>
<td>0</td>
</tr>
<tr>
<td>Invalid length:</td>
<td>0</td>
<td>Cache overflow:</td>
<td>0</td>
<td>Other errors:</td>
<td>0</td>
</tr>
<tr>
<td>Missing TLVs:</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TRANSMIT</th>
<th></th>
<th>Xmit fails:</th>
<th></th>
<th>No hostname:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Packet truncated:</td>
<td>0</td>
<td>Mem alloc fails:</td>
<td>0</td>
<td>Other errors:</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OTHER</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Init failures:</td>
<td>0</td>
</tr>
</tbody>
</table>

After the statistics are cleared, they begin to accumulate after the next CDP advertisement is sent or received.

### Using LLDP to detect network connectivity

Using LLDP to detect network connectivity consists of reviewing deployment considerations, enabling it on all ports, viewing neighbor devices, and adjusting LLDP configuration values as needed.

#### Before you begin

LLDP must also be enabled on any switches and routers before information about neighbor devices can be displayed.

#### About this task

ONTAP currently reports the following type-length-value structures (TLVs):

- Chassis ID
- Port ID
- Time-To-Live (TTL)
- System name
  The system name TLV is not sent on CNA devices.

Certain converged network adapters (CNAs), such as the X1143 adapter and the UTA2 onboard ports, contain offload support for LLDP:

- LLDP offload is used for Data Center Bridging (DCB).
- Displayed information might differ between the cluster and the switch.
  For example, the Chassis ID and Port ID data displayed by the switch might be different for CNA and non-CNA ports, but the data displayed by the cluster is consistent for these port types.

**Note:** The LLDP specification defines access to the collected information through an SNMP MIB. However, ONTAP does not currently support the LLDP MIB.

### Enabling or disabling LLDP

To discover and send advertisements to LLDP-compliant neighboring devices, LLDP must be enabled on each node of the cluster. By default, LLDP is disabled on all ports of a node.

#### About this task

The `lldp.enable` option controls whether LLDP is enabled or disabled on the ports of a node:
on enables LLDP on all ports.
off disables LLDP on all ports.

Steps

1. Display the current LLDP setting for a node, or for all nodes in a cluster:
   - Single node: run -node node_name options lldp.enable
   - All nodes: options lldp.enable

2. Enable or disable LLDP on all ports of a node, or on all ports of all nodes in a cluster:

<table>
<thead>
<tr>
<th>To enable or disable LLDP on...</th>
<th>Enter...</th>
</tr>
</thead>
<tbody>
<tr>
<td>A node</td>
<td>run -node node_name options lldp.enable {on</td>
</tr>
<tr>
<td>All nodes in a cluster</td>
<td>options lldp.enable {on</td>
</tr>
</tbody>
</table>

   - Single node:
     run -node node_name options lldp.enable {on|off}
   - All nodes:
     options lldp.enable {on|off}

Viewing LLDP neighbor information

You can view information about the neighboring devices that are connected to each port of the nodes of your cluster, provided that the port is connected to an LLDP-compliant device. You use the network device-discovery show command to view neighbor information.

Step

1. Display information about all LLDP-compliant devices that are connected to the ports on a node in the cluster:

   network device-discovery show -node node -protocol lldp

Example

The following command shows the neighbors that are connected to the ports on node cluster-1_01. The output lists the LLDP-enabled devices that are connected to each port of the specified node. If the -protocol option is omitted, the output also lists CDP-enabled devices.

```
cluster-1::> network device-discovery show -node cluster-1_01 -protocol lldp
Node/Local Port Device Interface Platform
---------- ------ ------------------------- ----------------  ----------------
cluster-1_01/lldp
e2a 0013.c31e.5c60            GigabitEthernet1/36
e2b 0013.c31e.5c60            GigabitEthernet1/35
e2c 0013.c31e.5c60            GigabitEthernet1/34
e2d 0013.c31e.5c60            GigabitEthernet1/33
```
Adjusting the interval for transmitting LLDP advertisements

LLDP advertisements are sent to LLDP neighbors at periodic intervals. You can increase or decrease the interval for sending LLDP advertisements depending on network traffic and changes in the network topology.

About this task

The default interval recommended by IEEE is 30 seconds, but you can enter a value from 5 seconds to 300 seconds.

Steps

1. Display the current LLDP advertisement time interval for a node, or for all nodes in a cluster:
   • Single node:
     run -node node_name options lldp.xmit.interval
   • All nodes:
     options lldp.xmit.interval

2. Adjust the interval for sending LLDP advertisements for all ports of a node, or for all ports of all nodes in a cluster:
   • Single node:
     run -node node_name options lldp.xmit.interval interval
   • All nodes:
     options lldp.xmit.interval interval

Adjusting the Time-To-Live value for LLDP advertisements

Time-To-Live (TTL) is the period of time for which LLDP advertisements are stored in cache in neighboring LLDP-compliant devices. TTL is advertised in each LLDP packet and is updated whenever an LLDP packet is received by a node. TTL can be modified in outgoing LLDP frames.

About this task

• TTL is a calculated value, the product of the transmit interval (lldp.xmit.interval) and the hold multiplier (lldp.xmit.hold) plus one.
• The default hold multiplier value is 4, but you can enter values ranging from 1 to 100.
• The default TTL is therefore 121 seconds, as recommended by IEEE, but by adjusting the transmit interval and hold multiplier values, you can specify a value for outgoing frames from 6 seconds to 30001 seconds.
• If an IP address is removed before the TTL expires, the LLDP information is cached until the TTL expires.

Steps

1. Display the current hold multiplier value for a node, or for all nodes in a cluster:
   • Single node:
     run -node node_name options lldp.xmit.hold
   • All nodes:
2. Adjust the hold multiplier value on all ports of a node, or on all ports of all nodes in a cluster:
   • Single node:
     `run -node node_name options lldp.xmit.hold hold_value`
   • All nodes:
     `options lldp.xmit.hold hold_value`
Copyright © 2019 NetApp, Inc. All rights reserved. Printed in the U.S.

No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

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.

Data contained herein pertains to a commercial item (as defined in FAR 2.101) and is proprietary to NetApp, Inc. The U.S. Government has a non-exclusive, non-transferrable, non-sublicensable, worldwide, limited irrevocable license to use the Data only in connection with and in support of the U.S. Government contract under which the Data was delivered. Except as provided herein, the Data may not be used, disclosed, reproduced, modified, performed, or displayed without the prior written approval of NetApp, Inc. United States Government license rights for the Department of Defense are limited to those rights identified in DFARS clause 252.227-7015(b).
Trademark

NETAPP, the NETAPP logo, and the marks listed on the NetApp Trademarks page are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.

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., 1395 Crossman Ave., Sunnyvale, CA 94089 U.S.
- Telephone: +1 (408) 822-6000
- Fax: +1 (408) 822-4501
- Support telephone: +1 (888) 463-8277
Index

10GbE connectivity
  converting 40GbE network cards for 39

A

about this guide
  deciding whether to use the Network Management Guide 7
active connections
  displaying 128
  displaying by client 128
  displaying by LIF 129
  displaying by protocol 129
  displaying by service 130
  displaying information about cluster 131
addresses
  changing subnet 62
admin SVMs
  introduction to managing the hosts table 85
advertisements
  enabling or disabling CDP to discover and send to neighboring devices 135
  enabling or disabling LLDP to discover and send to neighboring devices 139
  setting CDP interval 137
advertisements, CDP
  support for 134
advertisements, LLDP
  configuring TTL for 141
  setting transmit interval 141

B

balancing zones
  creating DNS load 86
broadcast domain
  configuration worksheet 11
  determining which ports are available 17
broadcast domain ports
  modifying MTU 36
broadcast domains
  adding ports to 49, 50
  changing MTU value for ports in 53
  creating 49
  creating a VLAN for maintaining separate, within the same network domain 35
  creating for the IPspace containing the SVM on which the CIFS server is to reside 20
  creating for the IPspace containing the SVM on which the CIFS server will reside 21
  defined 8
  deleting 54
  displaying 54
  displaying assigned IPspace 46
  example of using 48
  introduction to configuring for SVM traffic 48
  merging 52
  MTU size, setting the 49
  removing ports 52
  removing ports from 19, 50
  renaming 49
  splitting 52
  types of, defined 48

C

CDP
  clearing statistics 138
  considerations for using 134
  disabling 135
  enabling 135
  prerequisites for detecting network connectivity using 134
  setting advertisement interval 137
  showing current state 135
  support for 134
  viewing neighbor information 136
  viewing statistics 138
CDP messages
  configuring hold time for 137
CIFS servers
  configuring DNS services on the SVM 25
  configuring dynamic DNS on the SVM 27
  creating a broadcast domain for the IP space containing the SVM for 21
  creating an IPspace for the SVM for 17
  creating LIFs for 24
  creating SVM for hosting 22
  removing ports from existing broadcast domains 19
Cisco Discovery Protocol
  See CDP
Cisco Discovery Protocol (CDP)
  supported neighbor discovery protocols for
  displaying network connectivity 134
classification
  network traffic, DSCP marking 94
client applications
  displaying routing configuration information 102
client services
  displaying routing configuration information 102
clients
  displaying active connections to nodes by 128
cluster connections
  displaying 128
cluster LIFs
  firewall policy for 96
  role for 66
cluster management LIFs
  role for 66
cluster networking components
  defined 8
cluster ports
  displaying broadcast domain 54
  enabling or disabling CDP on 135
  viewing or clearing CDP statistics on 138
clusters
  displaying active connections by LIF 130
displaying active connections to nodes by client 128
displaying active connections to nodes by protocol 129
displaying active connections to nodes by service 130
displaying information about active connections in 131
displaying information about interface groups in 120
displaying information about LIFs in 121
displaying information about network ports in 118
displaying information about VLANs in 119
displaying IPspaces in 46
displaying listening connections in 132
displaying routing configuration information 102
enabling IPv6 on 92
enabling or disabling CDP on nodes in 135
enabling or disabling LLDP on nodes in 139
viewing information about CDP neighboring devices
connected to ports of nodes in 136
viewing information about LLDP neighboring
devices connected to ports of nodes in 140
when to use IPspaces to separate client traffic 42
commands
for diagnosing network problems 133
for managing Border Gateway Protocol (BGP) 82
for managing DNS host-name entries 85
for managing LIF failover groups and policies 59
for managing LIF service policies 72
for managing SNMP 112
snmp traps 106
comments
how to send feedback about documentation 145
communities
command for managing SNMP 112
compliance
disabling FIPS 90
enabling FIPS 89
compliance, FIPS
viewing status 90
configurations
displaying information about LIFs 121
introduction to viewing information about network 118
viewing DNS domain 124
configuring
DNS host-name entries, commands for 85
DNS host-name resolution, introduction to 83
DNS services on the SVM 25
dynamic DNS on the SVM 27
host-name resolution, introduction to 83
IPv6 addresses 92
QoS marking 94
SNMP, commands for 112
connections
active, displaying by client 128
active, displaying by LIF 130
active, displaying by protocol 129
active, displaying by service 130
active, displaying information about cluster 131
connections, cluster listening
displaying information about 132
connectivity, network
prerequisites for detecting with CDP 134
prerequisites for detecting with LLDP 139
creating
broadcast domains 49
failover groups 56
LIFs for CIFS servers 24
subnets 60
SVM for hosting CIFS server 22
VLANs 35
D
data LIFs
firewall policy for 96
role for 66
data ports
displaying broadcast domain 54
displaying LIF failover group 125
data SVMs
creating SNMP communities in 106
DDNS
introduction to 83
See also dynamic DNS
deleting
broadcast domains 54
IPspaces 46
subnets 64
devices, CDP-compliant
support for 134
devices, neighboring
viewing information about LLDP 140
differentiated services code point
See DSCP
disabling
FIPS 90
displaying
active connection information, in clusters 131
active connections by client 128
active connections by LIF 130
active connections by protocol 129
active connections by service 130
broadcast domains, ports, and MTU values in 54
cluster listening connections 132
DNS host name entries 124
interface group information 120
IPspaces 46
LIF failover group information 125
LIF failover policy information 125
LIF failover targets 126
LIFs in load balancing zones 127
network information, introduction to 118
network port information 118
QoS marking values for different protocols 95
route association to LIFs 122
static route information 122
subnets 63
VLAN information 119
distinct IP address spaces
when to use IPspaces to define 42
DNS
commands for managing host-name entries 85
configuration worksheet 11
configuring for host-name resolution using an
external DNS server 83
introduction to configuring for host-name resolution 83
DNS domains
  viewing configurations 124
DNS host name entries
  displaying 124
DNS hosts table
  displaying 124
DNS load balancing
  explained 86
  how it works 86
  supported protocols 86
DNS load balancing zones
  creating 86
DNS services
  configuring on the SVM 25
DNS zones
  defined 8
documentation
  how to receive automatic notification of changes to 145
  how to send feedback about 145
domain configurations
  viewing DNS 124
domain, broadcast
  configuration worksheet 11
domains
  changing MTU value for ports in broadcast 53
  deleting broadcast 54
  example of using broadcast 48
  introduction to configuring broadcast, for SVM traffic 48
  splitting broadcast 52
domains, broadcast
  creating 49
  merging 52
domains, network
  creating a VLAN for maintaining separate broadcast domains within the same 35
DSCP marking
  support for 94
duplicate IP addresses
  displaying LIF information to check for 121
dynamic DNS
  configuring on the SVM 27
Dynamic DNS
  See DDNS
dynamic routes
  removing from routing tables 104

F
failover
  disabling on a LIF 57
  enabling on a LIF 57
failover groups
  adding ports to LIF 56
  command for adding network ports to LIF 59
  command for deleting LIF 59
  command for displaying information about LIF 59
  command for removing network ports from LIF 59
  command for renaming LIF 59
  configuring settings for LIF 57
  creating LIF 56
  displaying information about LIF 125
  introduction to configuring for LIFs 56
failover policies
  configuring settings for LIF 57
  displaying type applied to LIF failover group 125
failover targets
  viewing LIF 126
failures
  introduction to configuring LIF failover groups to response to link 56
Federal Information Processing Standards
  disabling 90
  enabling 89
  introduction to configuring network security using 89
  viewing compliance status 90
feedback
  how to send comments about documentation 145
FIPS
  disabling 90
  enabling 89
  introduction to configuring network security using 89
  viewing compliance status 90
firewall policies
  commands for managing 100
  custom, creating and assigning to LIFs 98
  introduction to configuring for LIFs 96
  types of LIF 96
  valid values for LIF 96
  ways to manage 96
firewall services
  commands for managing 100
  introduction to configuring for LIFs 96
  ways to manage LIF 96

G
guides
  requirements for using the Network Management Guide 7

H
hold times
  configuring CDP message 137
home ports
  reverting LIFs to 78
host name entries
displaying DNS 124
host-name entries
commands for managing DNS 85
host-name lookup
introduction to configuring 83
using a local hosts file 83
using an external DNS server 83
host-name resolution
configuring DNS for, using an external DNS server 83
configuring the name service switch table for 84
introduction to configuring DNS for 83
introduction to managing the hosts table 85
hosts table
displaying DNS 124
hosts tables
introduction to managing 85

I
ifgrp
adding ports 33
implement LACP
to communicate group membership to the directly attached switch 31
information
how to send feedback about improving documentation 145
intercluster and management LIFs
set in different subnets 67
intercluster LIFs
firewall policy for 96
role for 66
interface group
characteristics of single-mode 29
load balancing 31, 32
interface group ports
modifying MTU settings for 37
interface groups
adding ports 33
characteristics of static multimode 29
creating 32
deleting 34
displaying information about 120
dynamic multimode 31
introduction to combining physical ports to create 28
load balancing 32
load balancing, IP address based 32
load balancing, MAC address based 32
modifying administrative settings 36
remove a port from 33
interfaces
logical, deleting 78
logical, reverting to home port 78
interfaces, logical
modifying 75
IP addresses
adding to a subnet 61
changing subnet range of 62
deleting a subnet to deallocate 64
displaying LIF information to check for duplicate 121
displaying subnet 63
introduction to configuring pools of, into subnets 60
removing from a subnet 61
when to use IPspaces to define distinct spaces 42
IP spaces
creating broadcast domains for 20
creating broadcast domains for the 21
IPspace
configuration worksheet 11
IPspaces
creating for SVMs 45
creating for the SVM on which the CIFS server will reside 17
default 44
defined 8
deleting 46
determining which ports are available to add to the new broadcast domain 17
displaying 46
displaying broadcast domains, ports, and MTU values in 54
displaying QoS marking values for different protocols for 95
displaying subnets in 63
every example of when to use for secure routing 42
explained 42
modifying the QoS marking values for 94
properties of 44
SVMs created by default 44
designed when to use to separate client traffic 42
IPv6
enabling on the cluster 92
enabling or disabling RA messages 93
IPv6 addresses
advantages of using 92
configuring 92

L
LACP detects
the inability of the node to communicate with the direct-attached switch port 31
the loss of link status 31
LIF
configuration worksheet 11
LIF failover
in NAS environments 10
LIFs
adding to or removing from load balancing zones 87
assigning service policy 71
associated with a static route 122
characteristics of, by role 67
command for configuring firewall settings 100
commands for managing failover groups and policies 59
configuring failover settings 57
creating 72
creating a service policy 69
creating an SNMP community and assigning it to 106
creating custom firewall policies and assigning to 98
creating failover group 56
creating for CIFS servers 24
defined 8, 65
deleting 78
disabling failover 57
displaying active connection information about 131
displaying active connections to nodes by 130
displaying cluster listening connections about 132
displaying failover group information 125
displaying failover targets 126
displaying in load balancing zones 127
displaying route association information about 102
enabling failover 57
how DNS load balancing works 86
introduction to configuring 65
introduction to configuring failover groups for 56
introduction to configuring subnets to make creation easier 60
migrating 76
modifying 75
port hierarchy of 65
ports that can host 65
reverting to home port 78
roles for 66
types of firewall policies for 96
viewing failover targets 126
viewing information about 121
link failures
introduction to configuring LIF failover groups to response to 56
Link Layer Discovery Protocol (LLDP)
supported neighbor discovery protocols for
displaying network connectivity 134
listening connections, cluster
displaying 132
LLDP
disabling 139
enabling 139
prerequisites for detecting network connectivity 139
showing current state 139
viewing neighbor information 140
LLDP advertisements
configuring TTL for 141
setting transmit interval 141
load balancing
about DNS 86
introduction to optimizing user traffic by 86
IP address based 31, 32
MAC address based 31, 32
methods, explained 86
multimode interface groups 31
sequential 31
load balancing zone
displaying LIFs in 127
load balancing zones
adding LIFs to 87
creating DNS 86
removing LIFs from 87
logical ports
defined 8
introduction to combining physical ports to create 28
management
network traffic, DSCP marking 94
Management Information Base
describes SNMP objects and traps 105
object identifiers 105
marking values, QoS
displaying 95
mask values
changing subnet 62
merging
broadcast domains 52
messages
configuring hold time for CDP 137
MIB
structure of management data 105
MIBs
describes SNMP objects and traps 105
object identifiers 105
migration
LIF, performing 76
modifying
MTU settings for interface group ports 37
network port attributes 36
monitoring
DNS domain configurations 124
DNS host name entries 124
interface groups 120
LIFs in load balancing zones 127
network information, introduction to 118
network ports 118
static routes 122
VLANs 119
MTU settings
modifying for interface group ports 37
MTU value
changing for ports in a broadcast domain 53
displaying for a broadcast domain 54
displaying for ports in a broadcast domain 54
MTU values
setting for ports in a broadcast domain 49
multimode interface groups
load balancing, IP address based 32
load balancing, MAC address based 32
load balancing, port-based 32
load balancing, round-robin 32
load balancing, sequential 32
multimode interface groups, static characteristics of 29
name resolution
introduction to configuring DNS for host 83
name service switch tables
configuring for host-name resolution 84
name services
configuration worksheet 11
name services switch
configuration worksheet 11
naming conventions
for network ports 28
NAS path failover
  workflow 10
neighbor discovery protocols
  types supported for displaying network connectivity 134
neighboring devices
  enabling or disabling CDP to discover and send
  advertisements to 135
  enabling or disabling LLDP to discover and send
  advertisements to 139
  viewing information about CDP 136
  viewing information about LLDP 140
network configuration
  introduction to viewing information about 118
network connectivity
  prerequisites for detecting with CDP 134
  prerequisites for detecting with LLDP 139
  supported neighbor discovery protocols for
  displaying 134
network domains
  creating a VLAN for maintaining separate broadcast
domains within the same 35
network interface cards
  converting 40GbE X1144A-R6 and 40GbE
  X91440A-R6 for 10GbE connectivity 39
network interfaces
  configuring virtual IP LIFs 79
  setting up BGP 79
  virtual IP 81
Network Management Guide
  requirements for using 7
network port attributes
  modifying 36
network ports
  creating LIF failover group of 56
  displaying information about 118
  displaying LIF information to verify subnet
  ownership 121
  introduction to configuring 28
  introduction to grouping into broadcast domains for
  SVM traffic 48
  monitoring health of 38
  removing a NIC from a node 40
  virtualized and physical types of 28
network problems
  commands for diagnosing 133
network security
  introduction to configuring using FIPS 89
network traffic
  classification of, DSCP marking 94
  introduction to balancing loads to optimize 86
  management of, DSCP marking 94
NFSv4 LIFs
  migrating 76
NICs
  removing from a node 40
node management LIFs
  role for 66
nodes
  considerations for enabling CDP on 134
  displaying active connections by client 128
  displaying active connections by LIF 130
  displaying active connections by protocol 129
  displaying active connections by service 130
  enabling or disabling CDP on cluster 135
  enabling or disabling LLDP on cluster 139
  migrating LIFs 76
  removing a NIC from 40
  viewing information about CDP neighboring devices
  connected to ports of 136
  viewing information about LLDP neighboring
devices connected to ports of 140
notifications
  configuring traphosts to receive SNMP 111
O
object identifiers
  and MIBs 105
OIDS
  and MIBs 105
P
physical ports
  default port roles 8
  defined 8
  introduction to combining to create interface groups
  28
  introduction to managing VLANs over 34
policies
  commands for managing firewall 100
  custom firewall, creating and assigning to LIFs 98
  displaying failover type applied to LIF failover
  group 125
  introduction to configuring LIF firewall 96
  types of LIF firewall 96
  ways to manage LIF firewall 96
pools
  IP address, introduction to configuring subnets of 60
port hierarchy
  of LIFs 65
port usage
  on a storage system 114
ports
  adding from interface groups 33
  adding to a broadcast domain 50
  changing MTU value of broadcast domain 53
  converting 40GbE network interface cards into
  multiple 10GbE 39
  creating interface groups 32
  creating LIF failover group of network 56
  description 8
  displaying active connection information about 131
  displaying assigned IPspace 46
  displaying broadcast domain 54
  displaying cluster listening connections about 132
  displaying information about network 118
  displaying interface group information 120
  displaying LIF failover group 125
  displaying LIF information to verify subnet
  ownership of network 121
  enabling or disabling CDP on 135
  enabling or disabling LLDP on 139
  internal and external 114
introduction managing VLANs over physical 34
introduction to configuring LIF failover groups for high availability 56
introduction to configuring network 28
migrating LIFs 76
moving from one broadcast domain into an existing broadcast domain 52
network, introduction to grouping into broadcast domains for SVM traffic 48
remove from an interface group 33
removing from a broadcast domain 50
removing from existing broadcast domains 19
that can host LIFs 65
viewing information about connected CDP neighboring devices 136
viewing information about connected LLDP neighboring devices 140
viewing or clearing CDP statistics on 138
ports, home
reverting LIFs to 78
ports, physical
introduction to combining to create interface groups 28
problems
commands for diagnosing network 133
protocol
service description 114
protocols
displaying active connection information about 131
displaying active connections to nodes by 129
displaying cluster listening connections for 132
displaying QoS marking values for 95
modifying the QoS marking values for 94
Q
QoS marking
configuring 94
support for 94
supported traffic types 94
QoS marking values
displaying 95
modifying 94
Quality of Service
See QoS
R
RA messages
enabling or disabling 93
reference guides
requirements for using the Network Management Guide 7
remote hosts
displaying active connection information about 131
resolution
introduction to configuring DNS for host-name 83
roles
for LIFs 66
round-robin load balancing
DNS, how it works 86
router-advertisement messages
enabling or disabling 93
routes
creating static 101
displaying information about static 122
static, deleting 102
routing
enabling multipath 101
example of when to use IPspaces for secure 42
introduction to managing in SVMs 101
routing tables, described 101
static routes, described 101
routing configurations
displaying information about cluster 102
routing groups
defined 8
routing tables
removing dynamic routes from 104
S
secure routing
example of when to use IPspaces for 42
servers, CIFS
configuring DNS services on the SVM 25
service
port description 114
services
commands for managing firewall 100
displaying active connection information about 131
displaying active connections by 130
displaying cluster listening connections for 132
introduction to configuring LIF firewall 96
ways to manage LIF firewall 96
Simple Network Management Protocol
See SNMP
SNMP
authKey security 109
authNoPriv security 109
authProtocol security 109
commands for managing 112
configuring traphosts to receive notifications 111
configuring v3 users 108
example 109
introduction to managing on the cluster 105
noAuthNoPriv security 109
security parameters 109
traps, types 106
SNMP community
creating and assigning to a LIF 106
SNMP traps
built-in 106
snmpwalk 109
static multimode interface groups
characteristics of 29
static routes
creating 101
deleting 102
described 101
displaying information about 122
statistics
viewing or clearing CDP 138
subnet
configuration worksheet 11
Index | 153

S

subnets
  adding IP addresses to 61
  changing address, mask value, or IP addresses of 62
  creating 60
defined 8
deleting 64
displaying 63
displaying LIF information to verify network port ownership 121
introduction to configuring 60
removing IP addresses from 61
renaming 60

SVM
  configuration worksheet 11
SVM management LIF
  firewall policy for 96

SVMs
  configuring dynamic DNS on 27
  creating broadcast domains for the IP spaces for the CIFS server 20, 21
  creating for CIFS server 22
  creating IPspaces 45
  creating IPspaces for the CIFS server 17
  creating LIFs for CIFS servers 24
  creating SNMP communities in data 106
displaying active connection information about 131
displaying active connections by LIF 130
displaying assigned IPspace 46
displaying information about associated LIFs 121
displaying route association to LIFs 102
displaying routing configuration information 102
displaying static route information 122
introduction to configuring LIF failover groups for high-availability connections to 56
introduction to managing routing in 101
introduction to managing SNMP on the cluster 105

SVMs, admin
  introduction to managing the hosts table 85

T

targets
  viewing LIF failover 126

Time-To-Live
  See TTL
traffic
  introduction to balancing network loads to optimize user 86
transmit intervals
  setting for LLDP advertisements 141

traphosts
  command for managing SNMP 112
  configuring to receive SNMP notifications 111

traps
  command for managing SNMP 112
  configuring traphosts to receive SNMP notifications 111

troubleshooting connectivity issues
  examining routing configurations 102

TTL
  configuring LLDP advertisements 141

Twitter
  how to receive automatic notification of documentation changes 145

U

UC compliance
  DSCP marking support for 94

unauthorized system access
  preventing by creating custom firewall policies and assigning to LIFs 98

Unified Capabilities
  See UC

user traffic
  introduction to balancing network loads to optimize 86

users
  command for managing SNMP 112

V

values, QoS marking
  displaying 95

viewing
  DNS domain configurations 124

virtual LANs
  displaying information about 119
  over physical ports, introduction to managing 34
  See also VLANs

VLANs
  creating 35
deleting 36
  displaying information about 119
  modifying MTU size 36
  over physical ports, introduction to managing 34
  restrictions when creating 35

W

workflow
  NAS path failover 10

WWPNs
  about configuring for LIFs 65

Z

zone load balancing
  DNS, how it works 86

zones
  creating DNS load balancing 86
  displaying LIFs in load balancing 127