Data ONTAP® 8.1
Network Management Guide For 7-Mode

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

Part number: 210-05511_A0
Updated for Data ONTAP 8.1.1 on 14 June 2012

Contents

Network interfaces on your storage system 10
Network interface naming 10
Maximum number of network interfaces 12
What the e0M interface is 12
Guidelines for configuring e0M 14
How to use the RLM or BMC to manage Data ONTAP remotely 14
Ways to configure the RLM 15
Ways to configure the BMC 15
How TSO increases outbound throughput 15
Data ONTAP support for TSO 16
Viewing the TSO statistics 16
What LRO is 17
Viewing the LRO statistics 17
Increasing storage availability by using ACP 18
Enabling ACP 18
Network considerations when using a Data ONTAP-v storage system 20
Standards and characteristics of Ethernet frames 21
What jumbo frames are 21
Network interface requirements for jumbo frames 22
Guidelines to configure clients for jumbo frames 22
Flow control 22
Network interface configuration 23
Configuring network interfaces 23
Configuring an IP address for a network interface 24
Specifying a subnet mask for a network interface 25
Specifying the prefix length for a network interface 26
Specifying a media type for a network interface 27
Specifying an MTU size for a network interface 27
Specifying the flow control type for a network interface 28
Specifying whether a network interface is trusted 29
Configuring a partner interface in an HA pair 30
Enabling or disabling negotiated failover for a network interface 31

Removing a primary IP address from a network interface 32
Specifying the number of DAD attempts 32
Creating or removing aliases 34
Changing the status of an interface 34
Blocking data traffic on network interfaces 35
Blocking data traffic on e0M 36
Restricting protocol access 37
Network interface information you can view 38
Viewing network interface settings 38
Viewing statistics of all active TCP connections 39
Viewing or clearing network interface statistics 40
Viewing statistics of dropped data packets 43
Support for IPv6 45
Ways to configure IPv6 addresses 45
IPv6 address types 45
IPv6 address scopes 45
IPv6 address states 46
How to transition from IPv4 to IPv6 46
Enabling or disabling IPv6 47
Types of address autoconfiguration 48
What stateless address autoconfiguration is 48
Enabling or disabling router-advertisement messages 48
What Neighbor Discovery does 49
ND message types 49
How DAD works with Data ONTAP 50
How routing in Data ONTAP works 53
How fast path works 53
Similarities and differences between fast path over IPv4 and IPv6 54
How to manage the routing table 54
What the routed daemon does 55
When the routed daemon should be turned off 55
How dynamic routing works for IPv6 56
Routing tables in a vFiler unit environment 56
Circumstances that might alter the routing table 56
Specifying the default route 57
How to enable or disable routing mechanisms 57

Enabling or disabling fast path 58
Enabling or disabling the routed daemon from the command-line interface 58
How to view the routing table and default route information 59
Viewing the routing table from the command-line interface 59
Viewing the default route information from the command-line interface 61
Modifying the routing table 61
How to maintain host-name information 64
How the /etc/hosts file works 64
Adding a host name in the /etc/hosts file 65
Hard limits for the /etc/hosts file 66
Changing the host name of a storage system 66
How to configure DNS to maintain host information 66
Configuring DNS from the command-line interface 67
How DNS resolves host names 68
DNS name caching 69
DNS information you can view 69
How to use dynamic DNS to update host information 70
How dynamic DNS updates work in Data ONTAP 70
Support for dynamic DNS updates in Data ONTAP 71
Enabling or disabling dynamic DNS updates 71
Disabling the transmission of DNS updates for an IP address 72
Changing the time-to-live setting for DNS entries 73
How to use NIS to maintain host information 73
How using NIS slaves can improve performance 74
How an NIS master is selected 74
Creating /etc/hosts from the NIS master 75
Guidelines for using NIS slaves 75
NIS administrative commands 76
How to configure NIS with Data ONTAP interfaces 76
Enabling NIS using the command-line interface 77
Specifying the NIS domain name 77
Specifying NIS servers to bind to your storage system 78
Enabling an NIS slave on your storage system 79
What NIS information you can view 79
Viewing NIS performance statistics 80

How VLANs work 82
How VLAN membership affects communication 83
GARP VLAN Registration Protocol 84
GVRP configuration for VLAN interfaces 84
VLAN tags 84
Advantages of VLANs 85
Prerequisites for setting up VLANs 86
Guidelines for setting up VLANs in Data ONTAP 86
The vlan command syntax 87
Creating a VLAN 88
Configuring a VLAN 89
IPv6 link-local addresses for VLANs 90
How to use VLANs for tagged and untagged network traffic 90
Adding an interface to a VLAN 91
Deleting VLANs 92
Enabling or disabling GVRP on your VLAN interface 92
Viewing VLAN statistics 93
Viewing statistics for a specific VLAN 94
How interface groups work in Data ONTAP 95
Types of interface groups 96
Single-mode interface group 96
Static multimode interface group 97
Dynamic multimode interface group 98
Load balancing in multimode interface groups 100
IP address and MAC address load balancing 100
Round-robin load balancing 100
Port-based load balancing 100
Guidelines for configuring interface groups 101
The ifgrp command 101
Creating a single-mode interface group 102
Selecting an active interface in a single-mode interface group 104
Designating a nonfavored interface in a single-mode interface group 105
Failure scenarios for a single-mode interface group 106
Creating a static multimode interface group 106
Creating a dynamic multimode interface group 107
Adding interfaces to an interface group 109

Deleting interfaces from an interface group 110
Viewing interface group status 111
What the interface group status information table contains 111
Viewing interface group statistics 113
Destroying an interface group 114
Second-level interface groups 114
Guidelines for creating a second-level interface group 115
Creating a second-level interface group 115
Enabling failover in a second-level interface group 116
Second-level interface groups in an HA pair 117
Creating a second-level interface group in an HA pair 119
How CDP works with Data ONTAP 121
Considerations for using CDP 121
Enabling or disabling CDP 122
Configuring hold time for CDP messages 122
Setting the intervals for sending CDP advertisements 123
Viewing or clearing CDP statistics 123
Viewing neighbor information by using CDP 125
How to monitor your storage system with SNMP 127
Types of SNMP traps 127
What MIBs are 128
What the SNMP agent does 129
How to configure the SNMP agent 129
Enabling or disabling SNMP 130
Configuring SNMPv3 users 130
Setting SNMP access privileges 131
Viewing or modifying your SNMP configuration 131
SNMP command syntax 132
SNMPv3 security parameters 133
Example: SNMP commands 134
User-defined SNMP traps 136
How SNMP traps work 136
How to define or modify a trap 136
Viewing or modifying trap values from the command-line interface 137
Example: Trap definitions 137
Command syntax for SNMP trap parameters 138

SNMP trap parameters 138
How to diagnose network problems 143
Diagnosing transport layer problems 144
Viewing diagnostic results 144
How to diagnose ping problems 145
Increasing the ping throttling threshold value 146
Checking the ping throttling threshold status 146
Disabling ping throttling 146
Protecting your storage system from forged ICMP redirect attacks 147
Improving TCP network congestion with Appropriate Byte Counting 148
Increasing the upper limit of initial window in TCP 148
Specifying the limit for increasing the congestion window in ABC 149
Network interface statistics 150
Statistics for Gigabit Ethernet controller VI, VII, and G20 interfaces 150
Statistics for Gigabit and 10 Gigabit Ethernet controllers T204, T210, and T320 interfaces 154
Statistics for the BGE 10/100/1000 Ethernet interface 157
Statistics for 10 Gigabit Ethernet Controller IX1 - SFP+ 161
Statistics for Dual 10G Ethernet Controller CNA - SFP+ 164
Statistics for Quad Gigabit Ethernet Controller 82850 167
Ways to improve your storage system's performance 171
IP port usage on a storage system 173
Host identification 173
/etc/services NNTP and TTCP ports 176
NFS-enabled ports 176
Ports not listed in /etc/services 177
FTP 177
SSH 178
Telnet 178
SMTP 179
Time service 179
DNS 179
DHCP 180
TFTP 180
HTTP 181

Kerberos 181
NFS 181
CIFS 182
SSL 182
SNMP 183
RSH 184
Syslog 184
The routed daemon 184
NDMP 185
SnapMirror and SnapVault 185
Error codes for the netdiag command 186
Copyright information 190
Trademark information 191
How to send your comments 192
Index 193

Network interfaces on your storage system

Your storage system supports physical network interfaces, such as Ethernet and Gigabit Ethernet interfaces, and virtual network interfaces, such as interface group and virtual local area network (VLAN). Each of these network interface types has its own naming convention.

Your storage system supports the following types of physical network interfaces:

- 10/100/1000 Ethernet
- Gigabit Ethernet (GbE)
- 10 Gigabit Ethernet

In addition, some storage system models have a physical network interface named e0M. It is a low-bandwidth interface of 100 Mbps and is used only for Data ONTAP management activities, such as running a Telnet, SSH, or RSH session.

Related concepts

Network interface configuration on page 23
How interface groups work in Data ONTAP on page 95
How VLANs work on page 82

Network interface naming

Network interface names are based on whether the interface is a physical or virtual network interface. Physical interfaces are assigned names based on the slot number of the adapter. Interface group names are user specified. VLANs are named by combining the interface name and VLAN ID.

Physical interfaces are automatically assigned names based on the slot where the network adapter is installed. Because physical interfaces are Ethernet interfaces, they are identified by a name consisting of "e," the slot number of the adapter, and the port on the adapter (if multi-port adapter). A multiport adapter has letters or numbers imprinted next to its ports.

- e<*slot_number*> if the adapter or slot has only one port
- e<*slot_number*><*port_letter*> if the adapter or slot has multiple ports

Interface group names are user specified. An interface group's name should meet the following criteria:

- It must begin with a letter.
- It must not contain any spaces.
- It must not contain more than 15 characters.
- It must not already be in use by another interface or interface group.

VLAN interface names are in the following format:

- *<physical_interface_name>-<vlan_ID>*
- *<ifgrp_name>-<vlan_ID>*

The following table lists interface types, interface name formats, and example of names that use these identifiers.

Interface type	Interface name format	Examples of names
Physical interface on a single-port adapter or slot	e*<slot_number>*	e0 e1
Physical interface on a multiple-port adapter or slot	e*<slot_number><port_letter>*	e0a e0b e0c e0d e1a e1b
Interface group	Any user-specified string that meets certain criteria	web_ifgrp ifgrp1
VLAN	*<physical_interface_name>-<vlan-ID>* or *<ifgrp_name>-<vlan_ID>*	e8-2 ifgrp1-3

Host names

When you run the `setup` command on a storage system for the first time, Data ONTAP creates a host name for each installed interface by appending the interface name to the host name of the storage system.

Note: The interface host names are not advertised by DDNS, but are available in the `/etc/hosts` file.

The following table shows examples of host names appended with the interface names.

Interface type	Host name
Single-port Ethernet interface in slot 0	toaster-e0
Quad-port Ethernet interface in slot 1	toaster-e1a toaster-e1b toaster-e1c toaster-e1d

Maximum number of network interfaces

Beginning with Data ONTAP 7.3, storage system can contain 256 to 1,024 network interfaces per system, depending on the storage system model, system memory, and whether they are in an HA pair.

The number of physical interfaces depends on the storage system model. Each storage system can support up to 16 interface groups.

You should run the `sysconfig` command and check the Memory size field displayed for the slot 0 system board of the storage system to determine the storage system memory.

The maximum number of network interfaces that each system can support is shown in the following table. The total number of interfaces can include physical, interface group, VLAN, vh, and loopback interfaces.

Storage system memory	Maximum number of network interfaces
2 GB or less	128
2 GB or less in an HA pair	256
6 GB or less	256
6 GB or less in an HA pair	512
More than 6 GB	512
More than 6 GB in an HA pair	1,024

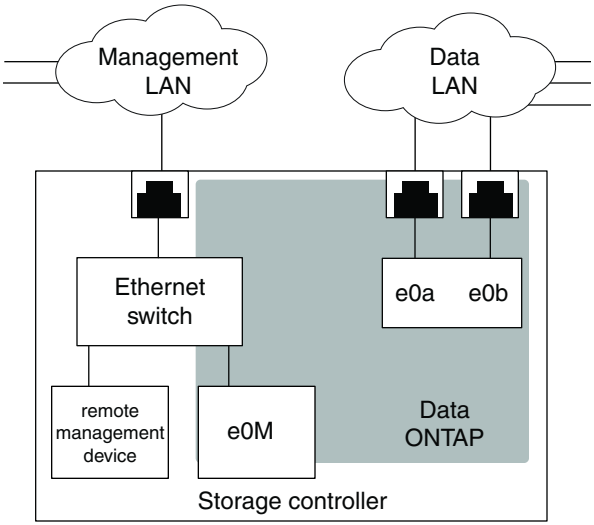
Related references

Network interface statistics on page 150

What the e0M interface is

Some storage system models have an interface named e0M. The e0M interface is dedicated to Data ONTAP management activities. It enables you to separate management traffic from data traffic on your storage system for security and throughput benefits.

On a storage system that has the e0M interface, the Ethernet port (indicated by a wrench icon on the rear of the chassis) is connected to an internal Ethernet switch. The internal Ethernet switch provides connectivity to the e0M interface and the remote management device, such as the SP, or the RLM. The following diagram illustrates the connections:



When you set up a system that includes the e0M interface, the Data ONTAP setup script recommends that you use the e0M as the preferred management interface for environments that use dedicated LANs to isolate management traffic from data traffic. The setup script then prompts you to configure e0M. The e0M configuration is separate from the configuration of the remote management device. Both configurations require unique IP addresses to allow the Ethernet switch to direct traffic to either the e0M interface or the remote management device. For information about how to set up the e0M interface, see the *Data ONTAP Software Setup Guide for 7-Mode*.

The e0M interface is a low-bandwidth, 100 BT (100 Mbps) interface. It does not support interface groups, VLANs, and jumbo frames.

After you set up the e0M interface, you can use it to access the storage system with the following management protocols, if they have been enabled:

- Telnet
- RSH
- HTTP or HTTPS
- SSH
- SNMP

Related concepts

Guidelines for configuring e0M on page 14

Related tasks

Blocking data traffic on e0M on page 36

Guidelines for configuring e0M

To achieve optimum storage performance in systems with e0M, you should ensure that e0M is not configured to serve high-bandwidth data traffic. Therefore, you must consider certain guidelines when configuring e0M.

The guidelines for configuring e0M are as follows:

- The e0M and high-bandwidth interfaces should not be configured in the same subnet.
- The e0M interface should be configured in a subnet that is not used for data traffic.
  If you cannot configure e0M in a dedicated management subnet, then you must disable e0M and ensure that system management activities are performed through a high-bandwidth interface.
- The e0M interface should not be configured with IP addresses that belong to the same subnet as the default route gateway.
- The IP addresses of the e0M interface should not be advertised by NIS or DNS to data clients.
- The e0M interface should be changed to `down` status by appending `ifconfig e0M down` to your `/etc/rc` file.

How to use the RLM or BMC to manage Data ONTAP remotely

You can manage your storage system locally from an Ethernet connection by using any network interface. However, to manage your storage system remotely, the system should have a Remote LAN Module (RLM) or Baseboard Management Controller (BMC). These provide remote platform management capabilities, including remote access, monitoring, troubleshooting, and alerting features.

If your data center configuration has management traffic and data traffic on separate networks, you can configure the RLM or the BMC on the management network.

With the RLM, you can remotely access the storage system in the following ways:

- Through the serial console
  The RLM is directly connected to the storage system through the serial console. You use the Data ONTAP CLI to administer the storage system and the RLM.
- Through an Ethernet connection using a secure shell client application
  You use the RLM CLI to monitor and troubleshoot the storage system.

With the BMC, you can access the storage system in the following ways:

- Through the serial console
  You use the Data ONTAP CLI to administer the storage system and the BMC.
- Through an Ethernet connection by using a secure shell client application
  You use the BMC CLI to monitor and troubleshoot the storage system.

For more information about the RLM and the BMC, see the *Data ONTAP System Administration Guide for 7-Mode*.

Ways to configure the RLM

Before using the RLM, you must configure it for your system and network. You can configure the RLM when setting up a new system with the RLM already installed, after setting up a new system with the RLM already installed, or when adding an RLM to an existing system.

You can configure the RLM by using one of the following methods:

- Initializing a storage system that has the RLM preinstalled
  When the storage system setup process is complete, the `rlm setup` command runs automatically. For more information about the entire setup process, see the *Data ONTAP Software Setup Guide for 7-Mode*.
- Running the Data ONTAP `setup` script
  The `setup` script ends by initiating the `rlm setup` command.
- Running the Data ONTAP `rlm setup` command
  For information about using the `rlm setup` command to configure the RLM, see the *Data ONTAP System Administration Guide for 7-Mode*.

When the `rlm setup` script is initiated, you are prompted to enter network and mail host information.

Ways to configure the BMC

Before using the BMC, you must configure it for your system and network. You can configure the BMC when setting up a new system that comes with the BMC. You can also configure the BMC after the initial system setup is complete.

You can configure the BMC by using one of the following methods:

- Initializing a storage system that has the BMC
  When the storage system setup process is complete, the `bmc setup` command runs automatically. For more information about the entire setup process, see the *Data ONTAP Software Setup Guide for 7-Mode*.
- Running the Data ONTAP `setup` script
  The `setup` script ends by initiating the `bmc setup` command.
- Running the Data ONTAP `bmc setup` command
  For information about using the `bmc setup` command to configure the BMC, see the *Data ONTAP System Administration Guide for 7-Mode*.

When the `bmc setup` script is initiated, you are prompted to enter network and mail host information.

How TSO increases outbound throughput

TCP segmentation offload (TSO), or Large send offload (LSO), reduces the CPU usage of the host system in high-bandwidth outbound network connections. In TSO, data segmentation is offloaded to

the NIC that divides the data into the default maximum transmission unit (MTU) size of the outgoing interface.

When a large amount of data is transmitted to a destination system, TCP in the host system segments the data into MTU sized packets. For each segment, a header buffer is allocated and a TCP header is inserted. The packets are sent down through the layers of the TCP stack, and each layer adds its own header to the packet. Processing all the created segments at each layer in the protocol stack can burden the CPU of the controller. This leads to the wastage of CPU cycles in the host system and causes an overhead to the host CPU. However, using TSO, this overhead can be passed down to the NIC to improve CPU efficiency of the host.

Data ONTAP support for TSO

Data ONTAP 8.0.1 and later releases support TSO. If fast path is enabled on the storage system, TSO is also enabled on all the network interfaces by default. TSO is also enabled on VLANs and any type of interface group if the underlying network interfaces support TSO.

TSO is automatically disabled under the following scenarios:

- In an interface group, if any of the constituent network interfaces do not support TSO
- If fast path is disabled
- In rate-limited connections such as SnapMirror with the throttle option enabled
  For more information about the `replication.throttle.enable` option, see the options(1) man page.

Viewing the TSO statistics

Viewing the TSO statistics enables you to find out whether TSO is functioning and gives you an estimate of the saved CPU cycles.

Step

1. To view the TSO statistics, enter the following command:

   `netstat -p tcp`

Example

The example shows a part of the `netstat -p tcp` command output.

```
49759833 segments sent using TSO
1637084830596 bytes sent using TSO
0 TSO segments truncated
0 TSO wrapped sequence space segments
```

The `netstat -p tcp` counters related to TSO are listed in the order of appearance in the output.

- Segments sent using TSO

  This counter indicates the number of segments sent by the application before the NIC performs TSO.

  For example, when 64 KB of data is transmitted, the TSO segments counter is incremented by 1.
- Bytes sent using TSO

  This counter indicates the number of bytes of segments sent by the application before the NIC performs TSO.

  For example, when 64 KB of data is transmitted, the TSO bytes counter is incremented by 64 KB.
- TSO segments truncated

  This counter indicates the number of times a segment has to be truncated because an application sends a buffer that is larger than the maximum size of the packet transmitted by NIC.

  For example, if the NIC supports a maximum segment size of 48 KB and an application sends 64 KB, then the buffer breaks the larger segment in two TSO segments. The first segment is 48 KB and the second segment is 16 (64 - 48) KB. This ensures that the NIC card does not transmit a large buffer.
- TSO wrapped sequence space segments

  This counter indicates the number of times a TSO segment wraps the 32-bit TCP sequence space.

What LRO is

Large Receive Offload (LRO), or Receive Side Coalescing (RSC), allows a network interface controller to combine incoming TCP/IP packets that belong to the same connection into one large receive segment before passing it to the operating system. It reduces CPU use because the TCP/IP stack is executed only once for a set of received Ethernet packets. LRO is available on some new network interface controllers. This feature is supported only on IPv4.

Viewing the LRO statistics

Viewing the LRO statistics enables you to find out whether, and to what extent, LRO is functioning.

Steps

1. Verify that LRO is supported and is operational on an interface by entering the following command:

   `ifconfig`

2. View the LRO statistics by entering the following command:

   `netstat -p tcp`

Examples

The following example shows a part of the `ifconfig` command output for e0c indicating that LRO is supported:

```
system1> ifconfig e0c
e0c: flags=0x1f4c867<UP,BROADCAST,RUNNING,MULTICAST,TCPCKSUM,LRO>
...
```

The following example shows a part of the `netstat -p tcp` command output:

```
system1> netstat -p tcp
...
6 segments received using LRO
183744 bytes received using LRO
...
```

The results in the example indicate the number of segments and bytes received by the application after the NIC performs LRO.

Increasing storage availability by using ACP

ACP, or Alternate Control Path, is a protocol that enables Data ONTAP to manage and control a SAS disk shelf storage subsystem. It uses a separate network (alternate path) from the data path, so management communication is not dependent on the data path being intact and available.

You do not need to actively manage the SAS disk shelf storage subsystem. Data ONTAP automatically monitors and manages the subsystem without operator intervention. However, you must provide the required physical connectivity and configuration parameters to enable the ACP functionality.

Note: You can install SAS disk shelves without configuring ACP. However, for maximum storage availability and stability, you should always have ACP configured and enabled.

After you enable ACP, you can use the `storage show acp` and `acpadmin list_all` commands to display information about your ACP subsystem.

Because ACP communication is on a separate network, it does not affect data access in any way.

Enabling ACP

ACP can increase your storage availability when you use SAS disk shelves. If your storage system model has a dedicated port for ACP, then ACP is enabled by default, and you do not need to explicitly enable ACP.

Before you begin

- Is the ACP subnet cabled on an isolated network, with no switches or hubs?
  For more information, see the *Installation and Service Guide* for your disk shelf.

- Have you identified a port that is not in use by any other subsystem?
- If you are configuring ACP for disk shelves attached to an HA pair, have you recorded the domain name and network mask to ensure that they are the same for both nodes?

About this task

The ACP subnet is a private Ethernet network that enables the ACP processor in the SAS module to communicate both with Data ONTAP and with the SAS IOMs in the disk shelves.

The ACP subnet is separate from the I/O data path that connects the disk shelves to the HBA on the storage controller. When you configure ACP on one of the system's network interfaces, you must supply a private domain name that conforms to the standard for private internet addresses (RFC1918). You can use the system default domain or another network name (that is, an IP address ending in 0) that conforms to the standard.

Steps

1. If your system does not have a dedicated port for ACP (e0p), ensure that the port you are assigning to ACP is not in use by any other subsystem by reviewing your `/etc/rc` file and entering the following command:

   `ifconfig` *`interface_name`*

   The interface you use for ACP should not be part of an Interface Group, and it should have no VLANs or IP addresses configured on it.

2. At the Data ONTAP command line, enter the following command:

   `acpadmin configure`

   If you have not previously configured the networking information for ACP, you are prompted for that information. When you select a domain name and network mask for the ACP interface, Data ONTAP automatically assigns IP addresses for the ACP interface on the storage controller and both I/O modules on each disk shelf on the ACP subnet.

3. You can verify your ACP connectivity by entering the following command:

   `storage show acp`

   The ACP Connectivity Status should show "Full Connectivity".

Example

For example, if you select e0P as the interface for ACP traffic, 192.168.0.0 as the ACP domain, and 255.255.252.0 as the network mask for the ACP subnet, the `storage show acp` command output looks similar to the following example:

```
my-sys-1> storage show acp

 Alternate Control Path:  enabled
 Ethernet Interface:      e0P
 ACP Status:              Active
```

```
 ACP IP address:          192.168.2.61
 ACP domain:              192.168.0.0
 ACP netmask:             255.255.252.0
 ACP Connectivity Status: Full Connectivity

Shelf Module      Reset Cnt    IP address      FW Version    Module
Type   Status
------------------------------------------------------------------------
-----------
7a.001.A          002          192.168.0.145    01.05
IOM6          active
7a.001.B          003          192.168.0.146    01.05
IOM6          active
7c.002.A          000          192.168.0.206    01.05
IOM6          active
7c.002.B          001          192.168.0.204    01.05
IOM6          active
```

Network considerations when using a Data ONTAP-v storage system

Data ONTAP-v storage systems support different network interfaces and fewer network options.

Because a Data ONTAP-v storage system is a software-based storage system, the physical network interfaces are provided by the server hosting the virtual machine. Access to Data ONTAP-v is across the server's network.

The following network interfaces are not available to Data ONTAP-v:

- The e0M interface is not available.
- RLM and BMC ports are not available for remote management.

The following items pertain to network software support when using platforms based on Data ONTAP-v technology:

- ACP is not supported.
- Jumbo frames are not supported.
- The underlying VMware ESX server only supports LACP in a static trunking configuration. The *VMware Networking Concepts* document (www.vmware.com/files/pdf/virtual_networking_concepts.pdf) contains additional information.
- The `cdpd` command will only display "neighbor" information for devices which support responding to CDP queries. The default VMware vSwitch supports advertising CDP to upstream Cisco devices, but does not support responding to information from Data ONTAP over the virtual network.

Standards and characteristics of Ethernet frames

Frame size and Maximum Transmission Unit (MTU) size are the two important characteristics of an Ethernet frame. The standard Ethernet (IEEE 802.3) frame size is 1,518 bytes. The MTU size specifies the maximum number of bytes of data that can be encapsulated in an Ethernet frame.

The frame size of a standard Ethernet frame (defined by RFC 894) is the sum of the Ethernet header (14 bytes), the payload (IP packet, usually 1,500 bytes), and the Frame Check Sequence (FCS) field (4 bytes). You can change the default frame size on Gigabit Ethernet network interfaces.

The MTU size specifies the maximum payload that can be encapsulated in an Ethernet frame. For example, the MTU size of a standard Ethernet frame is 1,500 bytes; this is the default for storage systems. However, a jumbo frame, with an MTU size of 9,000 bytes, can also be configured.

What jumbo frames are

Jumbo frames are larger than standard frames and require fewer frames. Therefore, you can reduce the CPU processing overhead by using jumbo frames with your network interfaces. Particularly, by using jumbo frames with a Gigabit or 10 Gigabit Ethernet infrastructure, you can significantly improve performance, depending on the network traffic.

Jumbo frames are packets that are longer than the standard Ethernet (IEEE 802.3) frame size of 1,518 bytes. The frame size definition for jumbo frames is vendor-specific because jumbo frames are not part of the IEEE standard. The most commonly used jumbo frame size is 9,018 bytes.

Jumbo frames can be used for all Gigabit and 10 Gigabit Ethernet interfaces that are supported on your storage system. The interfaces must be operating at or above 1,000 Mbps.

You can set up jumbo frames on your storage system in the following two ways:

- During initial setup, the `setup` command prompts you to configure jumbo frames if you have an interface that supports jumbo frames on your storage system.
- If your system is already running, you can enable jumbo frames by setting the MTU size on an interface.

Network interface requirements for jumbo frames

Before you enable jumbo frames on your storage system, jumbo frames must be enabled for the switch ports, client interfaces, and intermediate routers on the network. If your storage system and the client are on different subnets, the next-hop router must be configured for jumbo frames.

Guidelines to configure clients for jumbo frames

When configuring clients for jumbo frames, you should verify certain configurations, such as the TCP window size of the client, the MTU size of the client, storage system, and any intermediate subnet.

The guidelines for configuring clients for jumbo frames are as follows:

- The jumbo frames on the client and on your storage system should be configured.
  To find out how to configure jumbo frames on your client, check the network adapter documentation for your client.
- The TCP window size on the client should be enlarged depending on the MTU size.
  The minimum value for the client's window size should be two times the MTU size, minus 40, and the maximum value can be the highest value your system allows. Typically, the maximum value you can set for your client's TCP window is 65,535. If your storage system is configured to support jumbo frames and the client is not, the communication between the storage system and the client occurs at the client's frame size.
- The storage system and the UDP clients should be configured to have the same MTU size.
  UDP systems do not negotiate the MTU size. If your storage system and clients do not have the same MTU size, the storage system might send packets that the clients may not be able to receive.
- If the storage system and the client are on different subnets, then the MTU size of any intermediate subnets should be checked.
  If the storage system and the client (both configured to use jumbo frames) are on different subnets and an intermediate subnet does not support jumbo frames, the intermediate router fragments the IP packets as a result of which the advantages of using jumbo frames are lost.

Related tasks

Specifying an MTU size for a network interface on page 27

Flow control

Flow control enables you to manage the flow of frames between two directly connected link-partners. Flow control can reduce or eliminate dropped packets due to overrun.

To achieve flow control, you can specify a flow control option that causes packets called Pause frames to be used as needed. For example, link-partner A sends a Pause On frame to link-partner B when its receive buffers are nearly full. Link-partner B suspends transmission until it receives a Pause Off frame from link-partner A or a specified timeout threshold is reached.

Network interface configuration

Configuring network interfaces involves assigning IP addresses, setting network parameters and hardware-dependent values, specifying network interfaces, and viewing your storage system's network configuration.

When you configure network interfaces, you can do any or all of the following:

- Assign an IP address to a network interface.
- Set parameters such as network mask, broadcast address, and prefix length.

  Note: If IPv6 is enabled on your storage system, you can set only the prefix length. IPv6 does not have a network mask and does not support broadcast addresses.
- Set hardware-dependent values such as media type, MTU size, and flow control.
- Specify whether the interface should be attached to a network with firewall security protection.
- Specify whether the network interface must be registered with Windows Internet Name Services (WINS), if CIFS is running and at least one WINS server has been configured.
- Specify the IP address of an interface or specify the interface name on an HA pair partner for takeover mode.

  Note: When using IPv6 in an HA pair, you can specify only the partner interface name (and not the IP address) on the HA pair for takeover mode.
- View the current configuration of a specific interface or all interfaces that exist on your storage system.

Related concepts

Network interfaces on your storage system on page 10

Configuring network interfaces

You can configure network interfaces either during system setup or when the storage system is operating. When the storage system is operating, you can use the `ifconfig` command to assign or modify configuration values of your network interfaces.

During system setup, you can configure the IP addresses for the network interfaces. An `ifconfig` command is included in the `/etc/rc` file of the root volume for each network interface that you configured during the system setup. After your storage system has been set up, the `ifconfig` commands in the `/etc/rc` file are used to configure the network interfaces on subsequent storage system reboots.

You can use the `ifconfig` command to change values of parameters for a network interface when your storage system is operating. However, such changes are not automatically included in

the `/etc/rc` file. If you want your configuration modifications to be persistent after a reboot, you must include the `ifconfig` command values in the `/etc/rc` file.

Configuring an IP address for a network interface

You can configure IP addresses for your network interface during system setup. To configure the IP addresses later, you should use the `ifconfig` command. You can configure both IPv4 and IPv6 addresses for a network interface.

About this task

- Network configuration changes made by using the `ifconfig` command are not automatically included in the `/etc/rc` file. To make the configuration changes persistent after reboots, include the `ifconfig` command in the `/etc/rc` file.
- When you configure an IP address, your storage system creates a network mask based on the class of the address (Class A, B, C, or D) by default.

Step

1. To configure an IP address for a network interface, enter the following command:

   `ifconfig` *`interface_name IP_address`*

   `interface_name` is the name of the network interface.

   `IP_address` is the IP address that you want to assign to the network interface.

   #### Example

   To configure a quad-port Ethernet interface e3a to use the IPv4 address 192.0.2.10, enter the following command:

   `ifconfig e3a 192.0.2.10`

   To configure a quad-port Ethernet interface e3a to use the IPv6 address 2001:0db8:35ab:0:8a2e:0:0370:85, enter the following command:

   `ifconfig e3a 2001:0db8:35ab:0:8a2e:0:0370:85`

Related tasks

Specifying a subnet mask for a network interface on page 25

Specifying a subnet mask for a network interface

You must specify a subnet mask if you have created subnets that do not match the class boundary of the IPv4 address of the network interface. You can specify a subnet mask for a network interface by using the `ifconfig` command. IPv6 does not support subnet masks.

Before you begin

The IP address assigned to e0M must not belong to the same subnet as the IP address configured to serve data traffic.

Note: An EMS message is displayed if the management interface and network interfaces belong to the same subnet.

About this task

Data ONTAP allows you to configure a 32-bit subnet mask with all bits equal to 1.

Steps

1. To specify a subnet mask for an interface, enter the following command:

   ifconfig *interface_name* netmask *mask*

   interface_name is the name of the network interface.

   mask is the subnet mask.

   Example
   To configure a 24-bit mask for the interface e3a that you have already configured, enter the following command:

   ifconfig e3a netmask 255.255.255.0

2. To change the subnet mask for an interface that has been configured with a primary and an alias address, enter the following command for each IP address:

   ifconfig *interface_nameIP address* netmask *mask*

   Example
   To change the subnet mask of the primary IP address for the interface e3a, enter the following command:

   ifconfig e3a 172.25.206.110 netmask 255.255.255.0

   To change the subnet mask of the alias IP address for e3a, enter the following command:

   ifconfig e3a 120.120.1.1 netmask 255.255.255.0

Related concepts

Guidelines for configuring e0M on page 14

Related tasks

Configuring an IP address for a network interface on page 24

Specifying the prefix length for a network interface

Prefix length specifies the number of bits in the IP address that are to be used as the subnet mask. You can specify the prefix length for a network interface by using the `ifconfig` command.

About this task

For an IPv4 address, the prefix length must be less than or equal to 32 bits. For an IPv6 address, the prefix length must be less than or equal to 128 bits. The default value of the prefix length for an IPv6 address is 64 bits.

Step

1. To specify the prefix length, enter the following command:

   ifconfig *interface_name ip_address* prefixlen *length*

   ip_address is the IP address assigned to the network interface.

   length is the prefix length for the network interface.

   #### Example

   To configure a prefix length of 24 bits, enter the following command:

   ifconfig e0a 192.0.2.16 prefixlen 24

   To configure a prefix length of 64 bits for an IPv6 address, enter the following command:

   ifconfig e3a 2001:0db8:35ab:0:8a2e:0:0370:85 prefixlen 64

Specifying a broadcast address

You can use a broadcast address to send a message to all the machines on a subnet. You can specify a broadcast address by using the `ifconfig` command.

About this task

IPv6 does not support broadcast addresses.

Step

1. To specify a broadcast address, enter the following command:

   ifconfig *interface_name* broadcast *address*

   interface_name is the name of the network interface.

   address is the broadcast address.

Example

To set a broadcast address of 192.0.2.25 for the network 192.0.2.10 with subnet mask 255.255.255.0, enter the following command:

```
ifconfig e3a broadcast 192.0.2.25
```

Specifying a media type for a network interface

You can specify a media type for configuring the speed and duplex of a network interface by using the `ifconfig` command.

Step

1. To specify a media type, enter the following command:

   ```
   ifconfig interface_name mediatype type
   ```

   `interface_name` is the name of the network interface.

   `type` specifies the Ethernet media type used. The possible values are `tp`, `tp-fd`, `100tx`, `100tx-fd`, `auto`, or `10g-sr`.

   For more information, see the na_ifconfig(1) man page.

   Example

   To configure the interface e2a as a 100Base-TX full-duplex interface, enter the following command:

   ```
   ifconfig e2a mediatype 100tx-fd
   ```

Specifying an MTU size for a network interface

The maximum transmission unit (MTU) size is used to specify the jumbo frame size on 1 Gigabit Ethernet and 10 Gigabit interfaces. You can specify the MTU size for transmission between the storage system and its client by using the `ifconfig` command.

Step

1. To specify an MTU size, enter the following command:

   ```
   ifconfig interface_name mtusize size
   ```

   `interface_name` is the name of the network interface.

   `size` is the MTU to be used for the network interface.

   Example

   To specify an MTU size of 9000 for Gigabit Ethernet interface e8, enter the following command:

   ```
   ifconfig e8 mtusize 9000
   ```

Related concepts

Standards and characteristics of Ethernet frames on page 21

What jumbo frames are on page 21
Guidelines to configure clients for jumbo frames on page 22

Specifying the flow control type for a network interface

You can specify the flow control type for a network interface to manage the flow of frames between two directly connected link-partners. By controlling the transmission rate of data packets between the link-partners, you can effectively manage the network traffic.

Before you begin

- Depending on the network infrastructure, you must specify the flow control settings for a network interface.
  For example, if you set the flow control type on the network interface to `full`, the link-partner (switch) should be able to send and receive flow control frames too.
- The flow control settings of all the physical interfaces that constitute an interface group and VLANs must be the same.
- Flow control should normally be set to `none` for 10-GbE interfaces except for Converged Network Adapter (CNA) cards, where it cannot be disabled. If you disable flow control on the switch port, flow control is also disabled for devices such as CNAs.

About this task

The configured flow control setting is advertised during autonegotiation. If autonegotiation succeeds, the operational flow control setting is determined based on the negotiated speed and the value advertised by the other device. If autonegotiation fails, the configured flow control setting is used.

Note: Only 1-Gb interfaces support autonegotiation. For 10-Gb interfaces, the network interface on the storage system and the link-partner (usually a switch) should be configured with compatible settings.

Step

1. To specify the flow control type, enter the following command:

   `ifconfig` *`interface_name`* **`flowcontrol`** *`value`*

   `interface_name` is the name of the network interface.

   `value` is the flow control type. You can specify the following values for the `flowcontrol` option:

none	No flow control
receive	Able to receive flow control frames
send	Able to send flow control frames
full	Able to send and receive flow control frames

The default flow control type is `full`.

Example

To turn off flow control on the e0a interface, enter the following command:

```
ifconfig e0a flowcontrol none
```

Related concepts

Flow control on page 22

Specifying whether a network interface is trusted

You can specify whether a network interface is trustworthy or untrustworthy. When you specify an interface as untrusted (untrustworthy), any packets received on the interface are likely to be dropped. For example, if you run a `ping` command on an untrusted interface, the interface drops any ICMP response packet received.

About this task

Applications using protocols such as NFS, CIFS or HTTP can choose to accept packets only from trusted interfaces. If the destination interface is set as untrusted, it can receive packets from untrusted interfaces. Otherwise, the packets from untrusted interfaces are dropped. By default, only HTTP allows receiving packets from untrusted interfaces.

Step

1. To specify a network interface as trusted or untrusted, enter the following command:

   `ifconfig` *`interface_name`* `{trusted|untrusted}`

   `interface_name` is the name of the network interface.

   `trusted` specifies that the network interface is to be trusted.

   `untrusted` specifies that the network interface is not to be trusted.

   Example

   To specify that the network attached to interface e8 is not to be trusted for firewall security, enter the following command:

   ```
   ifconfig e8 untrusted
   ```

Configuring a partner interface in an HA pair

To prepare for a successful takeover in an HA configuration, you can map a network interface to an IP address or to another network interface on the partner node. During a takeover, the network interface on the surviving node assumes the identity of the partner interface.

Before you begin

When specifying the partner IP address, both the local network interface and the partner's network interface must be attached to the same network segment or network switch.

About this task

- If the network interface is an interface group, the partner interface must be denoted by an interface name and not an IP address.

  The partner interface can be an interface group or a physical network interface.
- You cannot specify the underlying physical ports of an interface group in a partner configuration.
- If IPv6 addresses are to be taken over, you must specify the partner interface, and not an IP address.

  Address to address mapping is not supported for IPv6 addresses.
- For the partner configuration to be persistent across reboots, you must include the `ifconfig` command in the `/etc/rc` file.

  For a successful takeover in both directions, you must repeat the partner configuration in the `/etc/rc` files of each node.
- When specifying the partner interface name, you can configure the interfaces symmetrically, for example map interface e1 on one node to interface e1 on the partner node.

  Though symmetrical configuration is not mandatory, it simplifies administration and troubleshooting tasks.

Step

1. Depending on the partner configuration that you want to specify, enter the following command:

If you want specify a...	Enter the following command...
Partner IP address	**ifconfig *interface_name* partner *address*** *interface_name* is the name of the network interface. *address* is the partner IP address.
Partner interface name	**ifconfig *interface_name* partner *partner_interface*** *partner_interface* is the name of the partner network interface.

Example: Specifying a partner IP address and partner interface name

Consider node1 and node2 are two storage systems in an HA configuration.

If the IP address of the interface e8 on node2 is 198.9.200.38, the following command allows the interface e1 of node1 to take over the IP address of node2 for the duration of the takeover:

```
node1> ifconfig e1 partner 198.9.200.38
```

Instead of specifying the IP address, you can also specify the partner interface name. The following command allows the interface e1 of node1 to assume the identity of e8 of node2 for the duration of the takeover:

```
node1> ifconfig e1 partner e8
```

Enabling or disabling negotiated failover for a network interface

You can enable or disable negotiated failover for a network interface to trigger automatic takeover if the interface experiences a persistent failure. You can use the `nfo` option of the `ifconfig` command to enable or disable negotiated failover.

About this task

You can specify the `nfo` option for an interface group. However, you cannot specify the `nfo` option for any underlying physical interface of the interface group.

Steps

1. To enable takeover during interface failure, enter the following command:

   `options cf.takeover.on_network_interface_failure on`

2. To enable or disable negotiated failover, enter the following command:

   `ifconfig` ***`interface_name`*** **`{nfo|-nfo}`**

   `interface_name` is the name of the network interface.

   `nfo` enables negotiated failover.

   `-nfo` disables negotiated failover.

   Example

   To enable negotiated failover on the interface e8 of an HA configuration, enter the following command:

   `ifconfig e8 nfo`

Removing a primary IP address from a network interface

You can remove a primary IP address from a network interface to disconnect the network interface from the network or reconfigure the network interface.

Before you begin

Ensure that you remove all the manually configured alias addresses for the interface.

Step

1. To remove a primary IP address, enter the following command:

   ifconfig *interface_name* 0

   interface_name is the name of the network interface.

   Alternatively, to remove a primary IPv4 address, you can use the following command:

   ifconfig *interface_name* 0.0.0.0

   Example

   To remove the primary address of the interface e3, enter the following command:

   ifconfig e3 0

   Note: To remove a primary IPv6 address, you can use either of these commands:

   - **ifconfig *interface_name* 0::0**
   - **ifconfig *interface_name* inet6 0**

Related tasks

Creating or removing aliases on page 34

Specifying the number of DAD attempts

To check the uniqueness of an IPv6 address, a node sends Neighbor Solicitation messages. You can use the `ifconfig` command to specify the number of consecutive Neighbor Solicitation messages (`dad_attempts`) to be sent while performing DAD on a tentative address before it can be configured.

Before you begin

IPv6 must be enabled on the storage system.

About this task

A value of zero for the `dad_attempts` option indicates that DAD is not performed on the tentative addresses. A value of one for the `dad_attempts` option indicates a single transmission with no follow-up retransmission and so on.

Step

1. Enter the following command:

   `ifconfig` *`interface_name`* `dad_attempts` *`value`*

   `interface_name` is the name of the interface

   `value` is the total number of consecutive Neighbor Solicitation messages sent while performing DAD on a tentative address. The default value is 2.

   You can set the `dad_attempts` *`value`* from 0 to 15 for physical interfaces and from 0 to 7 for interface groups and VLANs.

   Note: A `dad_attempts` *`value`* that is greater than 13 does not work in certain scenarios. Therefore, it is best to set the `dad_attempts` *`value`* to less than 13.

   #### Example

   You can configure the interface e0a for sending four consecutive Neighbor Solicitation messages by using the following command:

   `ifconfig e0a dad_attempts 4`

   The following is the output of the `ifconfig` command:

```
ifconfig e0a
e0a: flags=0x2d48867<UP,BROADCAST,RUNNING,MULTICAST,TCPCKSUM> mtu 1500
dad_attempts 4
inet6 fe80::2a0:98ff:fe06:c8f6 prefixlen 64 scopeid 0x3 autoconf
tentative
ether 00:a0:98:06:c8:f6 (auto-1000t-fd-up) flowcontrol full
```

Related concepts

How DAD works with Data ONTAP on page 50

Creating or removing aliases

You can create or remove an alias if you are changing the IP address of an interface. You should use the `alias` command to create an alias IP address, and use the `-alias` command to remove an alias IP address.

About this task

- The alias addresses are lost when the storage system reboots. If you want to make these changes persistent, include the `ifconfig` commands, which are used for configuring the alias addresses, in the `/etc/rc` file for the root volume.
- For IPv4 addresses, you can add an alias address only if a primary address for the interface exists.

  Note: For IPv6 addresses, the link-local and autoconfigured addresses are automatically added as alias addresses even without a primary address configured for an interface.

Step

1. To create or remove an alias, enter the following command:

   ifconfig *interface_name* [-]alias *address*

   #### Example

   The following example creates the alias IP address 192.0.2.30 for the interface e0 (already configured with IP address 192.0.2.21):

   ifconfig e0 alias 192.0.2.30

   The following example removes the 192.0.2.30 alias for the interface e0 specified in the previous example:

   ifconfig e0 -alias 192.0.2.30

Changing the status of an interface

You must make an interface inactive before performing tasks such as upgrading an interface, disabling a failed interface, or troubleshooting connectivity issues. You must again make the interface active after you complete the task. You can make an interface active or inactive by using the `ifconfig` command.

About this task

If you have enabled IPv6 on your storage system, you can bring up the interface without a valid address configured because a link-local address is created automatically for the interface. However, if the `/etc/rc` file contains an entry to keep a network interface in down status, a link-local address is not created.

Step

1. To change the status of an interface, enter the following command:

   ifconfig *interface* {up|down}

   up—makes the interface active

   down—makes the interface inactive

Blocking data traffic on network interfaces

For network security and efficient traffic management, you can block certain types of data traffic, including SnapMirror transfers, SnapVault transfers, and data transfers that use the CIFS, NFS, and NDMP protocols on selected or all network interfaces.

About this task

- If you block a data protocol on an interface by using the `interface.blocked.protocol` option, a client request using that particular data protocol on the interface fails. In such cases, use a different interface.
- You can block protocols on all interfaces, including interface groups and VLANs.
- Note that this topic does not apply to the management interface, e0M, which is handled differently.

Step

1. To block protocols on multiple network interfaces, enter the following command:

   options interface.blocked.*protocol_name interface_name,interface_name, ...interface_name*

   interface_name is the interface on which you want to block the protocol.

   protocol_name is the protocol that you want to block.

   To block multiple protocols on a single interface, you must repeat the command for each protocol.

Example

To block CIFS on the e9 interface, enter the following command:

options interface.blocked.cifs e9

To block NDMP on the e0a and e0b interfaces, enter the following command:

options interface.blocked.ndmp e0a,e0b

If you later want to unblock only the e0b interface, enter the command again, omitting e0b:

options interface.blocked.ndmp e0a

If you want to unblock a given protocol, for example, NDMP, from all the network interfaces, enter the following command with no space between the quotation marks:

options interface.blocked.ndmp ""

To block NDMP and NFS on the e0a interface, enter the following commands in succession:

options interface.blocked.ndmp e0a

options interface.blocked.nfs e0a

Related tasks

Blocking data traffic on e0M on page 36

Blocking data traffic on e0M

Data ONTAP 8.0.2 and later allows you to block data traffic that uses SnapMirror transfers, SnapVault transfers, and data transfers that use the CIFS, NFS, iSCSI, and NDMP protocols. Blocking high-bandwidth data traffic on e0M optimizes system performance.

Before you begin

In a single path SnapMirror transfer between the source and the destination, you can block SnapMirror traffic from e0M on the source storage system by using the `interface.blocked.mgmt_data_traffic` option; however, you cannot block SnapMirror traffic from e0M on the destination by using the `interface.blocked.mgmt_data_traffic` option. To block traffic on e0M, all the storage systems must use a consistent configuration. You must ensure that routes to the destination storage systems do not use e0M, and that neither SnapMirror nor NDMP is configured to use an IP address that is assigned to e0M on a destination storage system.

About this task

You can perform this task when you are upgrading to Data ONTAP 8.0.2 and later, where the default value of the `interface.blocked.mgmt_data_traffic` option is `off`. In new systems installed with Data ONTAP 8.0.2 and later, the default value of the `interface.blocked.mgmt_data_traffic` option is `on`.

Blocking of data traffic on e0M is supported over IPv6.

Step

1. To block data traffic on e0M, enter the following command:

   options interface.blocked.mgmt_data_traffic on

Restricting protocol access

If a protocol is enabled for Data ONTAP, you can restrict the protocol's access to the storage system by specifying the host name, IP address, or network interface name.

Step

1. At the storage system prompt, enter one of the following commands:

If you want to restrict a protocol's access to the storage system by using...	Enter...
host name or IP address	**options** *protocol*.**access host=[***hostname*|*IP_address*]**
network interface name	**options** *protocol*.**access if=***interface_name*

- *protocol* is the name of the protocol you want to allow access to the storage system. It can be **rsh**, **telnet**, **ssh**, **httpd**, **httpd.admin**, **snmp**, **ndmpd**, **snapmirror**, or **snapvault**.
- *hostname* is the name of the host to which you want to allow access by using *protocol*.
- *IP_address* is the IP address of the host to which you want to allow access by using *protocol*.
  The `ssh.access` and `rsh.access` options support both IPv4 and IPv6 addressing.
- *interface_name* is the network interface name of the host to which you want to allow access by using *protocol*.

  Note: If the `telnet.access` option is not set to `legacy`, the `trusted.hosts` option is ignored for Telnet. If the `httpd.admin.access` option is not set to `legacy`, the `trusted.hosts` option is ignored for `httpd.admin`. If the `snapmirror.access` option is not set to `legacy`, the `/etc/snapmirror.allow` file is ignored for SnapMirror destination checking.

For more information about controlling protocol access to a storage system by using multiple host names, IP addresses, and network interfaces, see the na_protocolaccess(8) man page.

For information about NDMP, see the *Data ONTAP Data Protection Tape Backup and Recovery Guide for 7-Mode*.

For information about SnapMirror or SnapVault, see the *Data ONTAP Data Protection Online Backup and Recovery Guide for 7-Mode*.

Related tasks

Setting SNMP access privileges on page 131

Network interface information you can view

You can view the status and performance statistics of your network interfaces, such as packets sent and received, cumulative or continuous packet traffic, collisions and other errors, active sockets, memory buffer, protocol-specific statistics, routing tables.

Data ONTAP provides the following commands that you can use to view network interface information:

Command	Information displayed
`ifconfig -a`	• Interface status (up or down) • Configuration parameters
`ifstat`	• Packets sent and received • Collisions and other errors • Negotiated media type settings between storage system interfaces and link partners
`netstat`	• Active sockets for each protocol • Memory buffer (mbuf) pool usage • Protocol-specific statistics for all protocols or a single protocol • Cumulative or continuous packet traffic for all interfaces or a single interface • Routing tables

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

Viewing network interface settings

To verify the network connectivity and diagnose any issues, you can view the network interface settings, such as interface status, IP address, and other network parameters. You can view the settings of all interfaces or a specific interface by using the `ifconfig` command.

Step

1. Depending on the network interface settings that you want to view, complete the following step:

If you want to view...	Enter the following command...
All interfaces	**`ifconfig -a`**

If you want to view...	Enter the following command...
A specific interface	**ifconfig *interface_name***

Viewing statistics of all active TCP connections

You can view the mapping network context of each TCP connection and the number of bytes of data sent and received over each TCP connection by using the `netstat` command.

Step

1. Depending on the statistics that you want to view, perform the following step:

If you want to view the...	Enter the following command...
Mapping context of each TCP connection	**netstat -aM**
Number of bytes of data sent and received over each TCP connection	**netstat -aB**

Example

The following example shows the output of the `netstat -aM` command:

```
system1> netstat -aM
Active TCP connections (including servers)
Ctx  Local Address         Remote Address         Swind Send-Q  Rwind Recv-Q State
 lg  *.443                 *.*                        0      0      0      0 LISTEN
 lg  *.22                  *.*                        0      0      0      0 LISTEN
 lg  *.10568               *.*                        0      0      0      0 LISTEN
 lg  *.10569               *.*                        0      0      0      0 LISTEN
 lg  *.10567               *.*                        0      0      0      0 LISTEN
 lg  *.10571               *.*                        0      0      0      0 LISTEN
 lg  *.8514                *.*                        0      0      0      0 LISTEN
 lg  *.514                 *.*                        0      0      0      0 LISTEN
 lg  *.23                  *.*                        0      0      0      0 LISTEN
 lg  *.8023                *.*                        0      0      0      0 LISTEN
 lg  *.4047                *.*                        0      0      0      0 LISTEN
 lg  *.4045                *.*                        0      0      0      0 LISTEN
 lg  *.4046                *.*                        0      0      0      0 LISTEN
 lg  *.2049                *.*                        0      0      0      0 LISTEN
 lg  *.111                 *.*                        0      0      0      0 LISTEN
 lg  *.28073               *.*                        0      0      0      0 LISTEN
 lg  *.32243               *.*                        0      0      0      0 LISTEN
 lg  *.22899               *.*                        0      0      0      0 LISTEN
  1  192.168.1.72.2049     192.168.1.36.800       33952    328  26280      0 ESTABLISHED
 lg  *.2049                *.*                        0      0      0      0 LISTEN

Active UDP sockets (including servers)
Local Address         Remote Address        Send-Q Recv-Q
*.10570               *.*                        0      0
*.69                  *.*                        0      0
*.161                 *.*                        0      0
*.4049                *.*                        0      0
*.4047                *.*                        0      0
*.4045                *.*                        0      0
*.4046                *.*                        0      0
*.2049                *.*                        0      0
*.111                 *.*                        0      0
```

```
*.21566                *.*                         0      0
*.520                  *.*                         0      0
```

The following example shows the output of the `netstat -aB` command:

```
netstat -aB
Active TCP connections (including servers)
Local Address         Remote Address         Swind Send-Q  Rwind Recv-Q State
Sent   Rcvd
localhost-20.1023     localhost-10.671       65535      0   8760      0 ESTABLISHED
0      0
localhost-20.8514     localhost-10.626       66608      1   8760      0 ESTABLISHED
1     44
localhost-20.18576    localhost-10.7951      66608      0   8760      0 ESTABLISHED
9284    606K
*.10568               *.*                        0      0      0      0 LISTEN
0      0
*.10569               *.*                        0      0      0      0 LISTEN
0      0
*.10567               *.*                        0      0      0      0 LISTEN
0      0
*.22                  *.*                        0      0      0      0 LISTEN
0      0
*.443                 *.*                        0      0      0      0 LISTEN
0      0
*.8514                *.*                        0      0      0      0 LISTEN
0      0
*.514                 *.*                        0      0      0      0 LISTEN
0      0
*.23                  *.*                        0      0      0      0 LISTEN
0      0
*.8023                *.*                        0      0      0      0 LISTEN
0      0
*.32243               *.*                        0      0      0      0 LISTEN
0      0
*.22899               *.*                        0      0      0      0 LISTEN
0      0

Active UDP sockets (including servers)
Local Address         Remote Address        Send-Q Recv-Q
*.10570               *.*                        0      0
*.69                  *.*                        0      0
*.161                 *.*                        0      0
```

Viewing or clearing network interface statistics

You can use the `ifstat` command to view the cumulative statistics of packets received and sent on a specified interface or on all interfaces. You can also use the `ifstat` command to clear the statistics.

About this task

- The `ifstat` command displays the cumulative network interface statistics that are gathered either from the time of the last reboot or from the last time you cleared them.
- If you use the `ifstat` command on a storage system that is part of an HA pair, the resulting information pertains only to the storage system on which the command was run. The information does not include statistics for the partner node in an HA pair.
- In an HA pair in takeover mode, the `ifstat` command displays the combined statistics of the packets processed by the network interface on the local node and those on the partner node.

  Because the statistics displayed by the `ifstat` command are cumulative, a giveback does not cause the statistics to zero out.

Step

1. Depending on the statistics that you want to view, perform the following step:

If you want to...	Enter the following command...
View the network interface statistics of all interfaces	**ifstat -a**
View the network interface statistics of a specific interface	**ifstat *interface_name*** *interface_name* is the name of the network interface.
Clear the network interface statistics of a network interface	**ifstat -z *interface_name***

The output of the `ifstat` command depends on the type of interface. For example, Ethernet or Gigabit Ethernet interfaces generate different types of statistics.

Example of showing the network interface statistics before and after clearing them

To view the statistics of the network interface e0a, enter the following command:

ifstat e0a

An output similar to the following is displayed.

```
system1> ifstat e0a

-- interface  e0a  (8 days, 20 hours, 10 minutes, 27 seconds) --

RECEIVE
 Frames/second:      13  | Bytes/second:      800  | Errors/
minute:       0
 Discards/minute:     0  | Total frames:      897k | Total bytes:
62415k
 Total errors:        0  | Total discards:      0  | Multi/
broadcast:   734k
 No buffers:          0  | Non-primary u/c:     0  | Tag
drop:           0
 Vlan tag drop:       0  | Vlan untag drop:     0  | CRC
errors:         0
 Runt frames:         0  | Fragment:            0  | Long
frames:         0
 Jabber:              0  | Alignment errors:    0  | Bus
overruns:       0
 Queue overflows:     0  | Xon:                 0  |
Xoff:               0
 Jumbo:               0  | Reset:               0  |
Reset1:             0
 Reset2:              0
TRANSMIT
 Frames/second:       2  | Bytes/second:      110  | Errors/
minute:       0
```

```
 Discards/minute:     0  | Total frames:      153k | Total bytes:
24129k
 Total errors:        0  | Total discards:      0  | Multi/
broadcast:  9478
 Queue overflows:     0  | No buffers:          0  | Max
collisions:      0
 Single collision:    0  | Multi collisions:    0  | Late
collisions:     0
 Timeout:             0  | Xon:                 0  |
Xoff:                0
 Jumbo:               0
LINK_INFO
 Current state:      up  | Up to downs:         0  |
Auto:               on
 Speed:           1000m  | Duplex:           full  |
Flowcontrol:       none
```

The following command clears and reinitializes the statistics for the network interface e0a:

ifstat -z e0a

The following sample output shows the network interface statistics for the network interface e0a immediately after the statistics are cleared.

```
system1> ifstat e0a

-- interface  e0a  (0 hours, 0 minutes, 8 seconds) --

RECEIVE
 Frames/second:       1  | Bytes/second:       32  | Errors/
minute:       0
 Discards/minute:     0  | Total frames:        7  | Total
bytes:       448
 Total errors:        0  | Total discards:      0  | Multi/
broadcast:     0
 No buffers:          0  | Non-primary u/c:     0  | Tag
drop:            0
 Vlan tag drop:       0  | Vlan untag drop:     0  | CRC
errors:          0
 Runt frames:         0  | Fragment:            0  | Long
frames:         0
 Jabber:              0  | Alignment errors:    0  | Bus
overruns:        0
 Queue overflows:     0  | Xon:                 0  |
Xoff:                0
 Jumbo:               0  | Reset:               0  |
Reset1:              0
 Reset2:              0
TRANSMIT
 Frames/second:       1  | Bytes/second:       17  | Errors/
minute:       0
 Discards/minute:     0  | Total frames:        4  | Total
bytes:       361
 Total errors:        0  | Total discards:      0  | Multi/
broadcast:     0
 Queue overflows:     0  | No buffers:          0  | Max
collisions:      0
 Single collision:    0  | Multi collisions:    0  | Late
```

```
collisions:     0
 Timeout:              0  | Xon:                  0  |
Xoff:                 0
 Jumbo:                0
LINK_INFO
 Current state:        up | Up to downs:          0  |
Auto:                 on
 Speed:             1000m | Duplex:             full |
Flowcontrol:        none
```

Related references

Statistics for Gigabit Ethernet controller VI, VII, and G20 interfaces on page 150
Statistics for the BGE 10/100/1000 Ethernet interface on page 157

Viewing statistics of dropped data packets

You can view the count of dropped data packets for each data protocol by using the `netstat -s` command. The count of dropped packets might increase when DNS advertises the management IP address to data clients, or data traffic is blocked on e0M, or misconfiguration of static route.

About this task

The number of filtered packets dropped for IP and UDP protocols are included in the weekly AutoSupport messages, included in the `netstat -s` command output.

Step

1. To view the count of filtered packets dropped for each protocol, enter the following command:

   `netstat -s`

Example

The example shows a part of the `netstat -s` command output.

```
system1> netstat -s

ip:
        4066350 total packets received
        0 bad header checksums
        0 with size smaller than minimum
        0 with size larger than maximum
        0 with data size< data length
        0 with header length< data size
        0 with data length< header length
        0 with bad options
        0 with incorrect version number
        0 packets with spoofed source address
        0 packets arrived on wrong port
        0 fragments received
        the last 2 src addr that send fragments:
```

```
        0.0.0.0 0.0.0.0
        0 fragments dropped (dup or out of space)
        0 malformed fragments dropped
        0 overlapping fragments discarded
        0 fragments dropped after timeout
        the last 2 src addr that have fragment time out:
        0.0.0.0 0.0.0.0
        0 packets dropped, too many fragments
        the last 2 src addr that sent too many fragments:
        0.0.0.0 0.0.0.0
        0 packets dropped, reassembly queue overflow
        0 packets reassembled ok
        3958788 packets for this host
        0 packets for unknown/unsupported protocol
        0 packets forwarded
        4986 packets not forwardable
        0 redirects sent
        3057658 packets sent from this host
        0 packets sent with fabricated ip header
        0 output packets dropped due to no bufs, etc.
        0 output packets discarded due to no route
        0 output datagrams fragmented
        0 fragments created
        0 datagrams that can't be fragmented
        498 filtered packets dropped
        0 packets dropped due to vfiler mismatch
        0 packets forwarded by source interface
```

Support for IPv6

Starting with Data ONTAP 7.3.1, Internet Protocol version 6 (IPv6) is supported on your storage system's network. IPv6 increases the IP address size from 32 bits (in IPv4) to 128 bits. This larger address space provides expanded routing and addressing capabilities.

Data ONTAP 7.3 and earlier used IPv4 for all the addressing and networking requirements. However, IPv4 has many limitations, such as limited address space and security. To address these limitations, the Internet Engineering Task Force (IETF) developed a new version of IP, called IPv6.

You can enable the IPv6 option and configure IPv6 addresses on the network interfaces of the storage system. IPv6 addresses can also be automatically configured.

Ways to configure IPv6 addresses

IPv6 addresses can be configured on the network interfaces of your storage system, either manually or automatically. The configuration of an IPv6 address depends on the type and scope of the address.

IPv6 address types

There are three types of IPv6 addresses: unicast, anycast, and multicast.

Unicast address	This address identifies a single interface. A data packet sent to a unicast address is delivered only to the interface that is identified by that address.
Anycast address	This address identifies a set of interfaces. A data packet sent to an anycast address is delivered to the nearest interface (according to the routing protocols' measure of distance) that is identified by that address. **Note:** Anycast address is not supported in Data ONTAP.
Multicast address	This address identifies a set of interfaces. A data packet sent to a multicast address is delivered to all the interfaces that are identified by that address. **Note:** In IPv6, multicast addresses replace broadcast addresses.

IPv6 address scopes

IPv6 addresses fall under three scopes: global, link-local, and unique local.

Global address	This address has an unlimited scope.
Link-local	This address has a link-only scope that can be used to reach neighboring nodes that are attached to the same link. This address is automatically assigned to a network interface.

Note: FreeBSD user applications like NTP, ASUP, and NDMP cannot communicate with destinations that have IPv6 link- local addresses.

Unique local address	The address scope is limited to a local site or local set of sites. These addresses cannot be routed on the global Internet.

IPv6 address states

Before and after an IPv6 address is assigned, it goes through various states, such as tentative address, duplicate address, and preferred address. These address states are applicable to both manually and automatically configured addresses.

An IPv6 address can have one or more of the following states:

Tentative address	An address whose uniqueness on a link is being verified. When an address is configured on a network interface (either manually or automatically), the address is initially in the tentative state. Such an address is not considered to be assigned to an interface. An interface discards received packets addressed to a tentative address, but accepts Neighbor Discovery packets related to Duplicate Address Detection (DAD) for the tentative address.
Duplicate address	If DAD finds that an address is not unique, it is moved to the duplicate state. Such an address cannot be used for sending and receiving data.
Preferred address	An address used to send and receive data packets from and to a network interface without any restriction on the upper layer protocols.
Deprecated address	A preferred address becomes deprecated when its preferred lifetime expires. The use of this address is discouraged, but not prohibited.
Valid address	A uniquely verified address that you can assign to a network interface for sending and receiving data. A valid address can be a preferred or deprecated address.
Invalid address	A network interface address that can no longer send or receive data packets. A valid address becomes invalid when its valid lifetime expires. An invalid address is removed from the network interface.

How to transition from IPv4 to IPv6

A transition mechanism enables IPv6 hosts and routers to be compatible with IPv4 hosts and routers. Starting with Data ONTAP 7.3.1, a dual stack mechanism is used for transitioning from IPv4 to IPv6.

In the dual stack mechanism, the following modes are supported:

- Only IPv4 mode: In this mode, IPv6 is not enabled.
- Only IPv6 mode: In this mode, IPv6 is enabled and IPv4 addresses are not configured on any interface.

- IPv6/IPv4 mode: In this mode, IPv6 is enabled and both IPv4 and IPv6 addresses are configured on the network interfaces.

  Attention: In the "Only IPv6 mode," address lookup can return both IPv4 and IPv6 addresses. If you use an IPv4 address to set up communication in the "Only IPv6 mode," the communication fails. Therefore, you should have at least one IPv4 address configured in a network interface and then use the "IPv6/IPv4 mode."

Data ONTAP does not support the following IPv6 transition mechanisms (defined in RFC 2893):

- Configured tunneling of IPv6 over IPv4
- IPv4-mapped IPv6 addresses
- Automatic tunneling of IPv6 over IPv4

Enabling or disabling IPv6

You can enable IPv6 on all the interfaces of your storage system either during setup or when the storage system is in operation. You can disable IPv6 on your storage system if you want to revert to IPv4 addressing.

About this task

- You can enable IPv6 during initial system configuration when the `setup` command is run for the first time. If you want to enable IPv6 later, you can rerun the `setup` command or configure IPv6 manually. For more information about the `setup` command, see the *Data ONTAP Software Setup Guide for 7-Mode*.
- You can enable IPv6 only for the entire storage system, but not for a network interface or a vFiler unit.

Step

1. To enable or disable IPv6 when the storage system is in operation (not during setup), enter the following command:

   `options ip.v6.enable {on|off}`

   `on`—Enables IPv6

   `off`—Disables IPv6

After you finish

If you have enabled IPv6 when the storage system is in operation, you must manually restart all server applications, except CIFS, FTP, and HTTP, to run over IPv6. For CIFS, FTP, and HTTP to work over IPv6, you must enable their individual IPv6 options. For more information about the protocols supported over IPv6, see the *Data ONTAP File Access and Protocols Management Guide for 7-Mode*.

Note: If the applications are running only over IPv4, you do not need to restart the applications.

Types of address autoconfiguration

IPv6 defines both a stateful and a stateless address autoconfiguration mechanism. Data ONTAP 7.3.1 and later supports IPv6 stateless address autoconfiguration.

The Neighbor Discovery protocol is one of the protocols that facilitates address autoconfiguration.

Related concepts

What Neighbor Discovery does on page 49

What stateless address autoconfiguration is

The stateless address autoconfiguration mechanism allows a host to generate its own addresses by using a combination of locally available information and router-advertised information. The stateless address autoconfiguration requires minimal manual configuration of hosts and routers.

Data ONTAP supports the following two types of autoconfigured IPv6 addresses:

- Autoconfigured address based on the router-advertised prefix: This address is a combination of the network prefix, which is router-advertised, and the network interface identifier.
- Autoconfigured link-local address: In the absence of routers, a host can generate only link-local addresses. Link-local addresses allow communication between hosts and routers that are on the same link.

RFC 2462 describes address autoconfiguration.

Enabling or disabling router-advertisement messages

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. The prefix information option in a RA message is used by the storage system to configure routes or autoconfigure IPv6 addresses as part of stateless autoconfiguration.

About this task

- By default, the option is set to on.
- If you do not want the MAC address of the network interfaces to be viewed by any network for security reasons, you can disable the RA address autoconfiguration.
- Disabling the RA option does not remove the existing autoconfigured addresses and the existing routes.
- When the RA option is disabled, the RA message is dropped.

  Therefore, default route is not learned dynamically, the default router failover is disabled, and link MTU updates stop. You can still configure default routes statically.
- You should not use autoconfigured addresses for configuring any vFiler unit.

However, if you want to assign any autoconfigured address to a vFiler unit, then you must disable the RA address autoconfiguration prior to address assignment.

- You should not use the IPv6 prefix advertised by the local router for static configuration.
  You should not statically configure the non-default vFiler units with the same IPv6 addresses that are generated by the autoconfiguration.

Step

1. To enable or disable RA address autoconfiguration, enter the following command:

   `options ip.v6.ra_enable {on|off}`

   `on`—Accepts the RA messages

   `off`—Rejects or drops the RA messages

What Neighbor Discovery does

The Neighbor Discovery (ND) protocol enables hosts and routers to discover the presence of neighboring IPv6 hosts and routers. The ND protocol also helps in identifying the link-layer address of hosts and routers and in performing Duplicate Address Detection (DAD).

The ND protocol replaces the IPv4 protocols, such as Address Resolution Protocol (ARP), Internet Control Message Protocol (ICMP) Router Discovery, and ICMP Redirect.

The various ND mechanisms for enabling interaction between nodes that are on the same link, as described in RFC 2461.

ND message types

There are five types of ND messages: Router Solicitation, Router Advertisement, Neighbor Solicitation, Neighbor Advertisement, and Redirect. You can specify various ND options in an ND message.

Router Advertisement and Router Solicitation messages facilitate host-router discovery functions. Neighbor Solicitation and Neighbor Advertisement messages facilitate exchange of information between neighboring hosts on the same network. The Redirect message is used to inform a host of a better route for sending data packets to a particular destination. All the ND message types use the Internet Control Message Protocol version 6 (ICMPv6) message structure.

The ND options that can be specified in an ND message are the source link-layer address, target link-layer address, prefix information, MTU, and redirected header. These ND options provide additional information such as MAC addresses, on-link network prefixes, on-link MTU information, and redirection data.

Note: Data ONTAP supports a maximum of 10 options in an ND message.

How DAD works with Data ONTAP

Before assigning unicast addresses to an interface, Duplicate Address Detection (DAD) is performed on the addresses to ensure that the addresses configured on a link are unique. DAD is performed on all unicast addresses (both manually and automatically configured). When the DAD procedure fails for an address, the address is not configured.

DAD prevents multiple nodes from using the same address simultaneously. DAD is performed on all unicast addresses of a network interface, provided the value of the `dad_attempts` option for that interface is greater than zero.

To check the uniqueness of an address, a node sends Neighbor Solicitation messages, each separated by an interval of 1 second. The number of Neighbor Solicitation messages sent is equal to the value of the `dad_attempts` option for the network interface.

An address on which the DAD procedure is applied remains in the tentative state until the procedure has been successfully completed. The target address of the Neighbor Solicitation message is set to the address that is being checked and remains in the tentative state. If the node receives a valid Neighbor Advertisement message with the tentative address as target, the tentative address is not unique. The tentative address is marked duplicated and cannot be used for any data communication.

If DAD fails for a link-local address, the network interface is configured to the `down` status.

If a node does not receive a Neighbor Advertisement message after sending the Neighbor Solicitation messages for a tentative address, the address is considered unique. When an address is determined to be unique, it is assigned to the network interface.

Example: Duplicated unicast address

The following example shows DAD failure for a unicast address, where the address state changes from tentative to duplicated.

```
system1> ifconfig e0b 2001:0db8::99
system1> ifconfig e0b
e0b: flags=0x2508866<BROADCAST,RUNNING,MULTICAST,TCPCKSUM> mtu 1500
inet6 2001:0db8::99 prefixlen 64 tentative
ether 00:a0:98:08:64:07 (auto-1000t-fd-cfg_down) flowcontrol full
```

```
system1> Wed Aug 6 09:24:44 GMT [system1:netif.linkUp:info]: Ethernet
e0b: Link up.
Wed Aug 6 09:24:44 GMT [system1:netinet6.nbr.dad.dtcDupAdr:error]:
e0b: DAD detected duplicate IPv6
address 2001:0db8::99: %d NS, 0 NA.
Wed Aug 6 09:24:44 GMT [system1:netinet6.nbr.dad.complete:error]:
e0b: DAD complete for 2001:0db8::99- duplicate found.
Wed Aug 6 09:24:44 GMT [system1:netinet6.nbr.manl.intvtnReq:error]:
e0b: Manual intervention required.
Wed Aug 6 09:24:45 GMT [system1:netinet6.nbr.dadStrc.notFnd1:error]:
nd6_dad_timer: DAD structure is not found.
system1> ifconfig e0b
```

```
e0b: flags=0x2d48867<UP,BROADCAST,RUNNING,MULTICAST,TCPCKSUM,LINK_UP>
mtu 1500
inet6 2001:0db8::99 prefixlen 64 duplicated
inet6 fe80::2a0:98ff:fe08:6407 prefixlen 64 scopeid 0x2 autoconf
ether 00:a0:98:08:64:07 (auto-1000t-fd-up) flowcontrol full
```

Example: Duplicated link-local address

The following example shows DAD failure for a link-local address, where the network interface is configured to the down status.

```
system1> ifconfig e0b up
system1> Tue Jul 22 16:46:38 GMT [system1: netif.linkUp:info]:
Ethernet e0b: Link up.
Tue Jul 22 16:46:39 GMT [system1: netinet6.nbr.dad.dtcDupAdr:error]:
e0b: DAD detected duplicate IPv6 address
fe80:0002::02a0:98ff:fe08:6407: %d NS, 0 NA.
Tue Jul 22 16:46:39 GMT [system1: netinet6.nbr.dad.complete:error]:
e0b: DAD complete for fe80:0002::02a0:98ff:fe08:6407 - duplicate
found.
Tue Jul 22 16:46:39 GMT [system1: netinet6.nbr.manl.intvtnReq:error]:
e0b: Manual intervention required.
Tue Jul 22 16:46:39 GMT [system1: netif.linkInfo:info]: Ethernet e0b:
Link configured down.
Tue Jul 22 16:46:40 GMT [system1:
netinet6.nbr.dadStrc.notFnd1:error]: nd6_dad_timer: DAD structure is
not found.
```

```
system1> ifconfig -a
e0a: flags=0x2508866<BROADCAST,RUNNING,MULTICAST,TCPCKSUM> mtu 1500
        ether 00:a0:98:08:64:06 (auto-1000t-fd-cfg_down) flowcontrol
full
e0b: flags=0x2508866<BROADCAST,RUNNING,MULTICAST,TCPCKSUM> mtu 1500
        ether 00:a0:98:08:64:07 (auto-1000t-fd-cfg_down) flowcontrol
full
e0c: flags=0x2508866<BROADCAST,RUNNING,MULTICAST,TCPCKSUM> mtu 1500
        ether 00:a0:98:08:64:08 (auto-1000t-fd-cfg_down) flowcontrol
full
e0d: flags=0x2508866<BROADCAST,RUNNING,MULTICAST,TCPCKSUM> mtu 1500
        ether 00:a0:98:08:64:09 (auto-1000t-fd-cfg_down) flowcontrol
full
e0e: flags=0x2508866<BROADCAST,RUNNING,MULTICAST,TCPCKSUM> mtu 1500
        ether 00:a0:98:08:64:0a (auto-1000t-fd-cfg_down) flowcontrol
full
e0f: flags=0x2508866<BROADCAST,RUNNING,MULTICAST,TCPCKSUM> mtu 1500
        ether 00:a0:98:08:64:0b (auto-1000t-fd-cfg_down) flowcontrol
full
lo:
flags=0x1948049<UP,LOOPBACK,RUNNING,MULTICAST,TCPCKSUM,LINK_UP,UDPCKSU
M> mtu 8160
        inet 127.0.0.1 netmask 0xff000000 broadcast 127.0.0.1
        inet6 fe80::1 prefixlen 64 scopeid 0x7 autoconf
```

```
        inet6 ::1 prefixlen 128
        ether 00:00:00:00:00:00 (VIA Provider)
```

Related tasks

Specifying the number of DAD attempts on page 32

How routing in Data ONTAP works

You can have Data ONTAP route its own outbound packets to network interfaces. Although your storage system can have multiple network interfaces, it does not function as a router. However, it can route its outbound packets.

Data ONTAP uses two routing mechanisms:

Fast path	Data ONTAP uses this mechanism to route NFS packets over UDP and to route all TCP traffic. By default, fast path is enabled on the storage system.
Routing table	To route IP traffic that does not use fast path, Data ONTAP uses the information available in the local routing table. The routing table contains the routes that have been established and are currently in use, as well as the default route specification.

How fast path works

Fast path is an alternative routing mechanism to the routing table. In fast path, the responses to incoming network traffic are sent back by using the same interface as the incoming traffic. By avoiding the routing table lookup, fast path provides a quick access to data.

If fast path is enabled on an interface group and a physical interface in that group receives an incoming request, the same physical interface might not send a response to the request. Instead, any other physical interface in an interface group can send the response.

How fast path works with NFS/UDP

NFS/UDP traffic uses fast path only when sending a reply to a request. The reply packet is sent out on the same network interface that received the request packet.

For example, a storage system named toaster uses the toaster-e1 interface to send reply packets in response to NFS/UDP requests received on the toaster-e1 interface.

Fast path is used only in NFS/UDP. However, fast path is not used in other UDP-based NFS services such as portmapper, mountd, and nlm.

How fast path works with TCP

In a TCP connection, fast path is disabled on the third retransmission and the consecutive retransmissions of the same data packet. If Data ONTAP initiates a connection, Data ONTAP can use fast path on every TCP packet transmitted, except the first SYN packet. The network interface that is used to transmit a packet is the same interface that received the last packet.

Fast path not compatible with asymmetric routing

In a symmetric network, the destination MAC address of the response packet is that of the router that forwarded the incoming packet. However, in asymmetric networks, the router that forwards packets

to your storage system is not the router that forwards packets sent by the storage system. Therefore, in asymmetric networks, you must disable fast path.

Related concepts

Data ONTAP support for TSO on page 16

Related tasks

Enabling or disabling fast path on page 58

Similarities and differences between fast path over IPv4 and IPv6

Starting with Data ONTAP 7.3.3, fast path is supported over IPv6. Fast path over IPv4 and IPv6 provide improved storage system performance. However, fast path over IPv6 does not provide load balancing between multiple interfaces like IPv4 does.

Similarities between fast path over IPv4 and IPv6

Fast path over IPv4 and IPv6 provide improved system performance because of the following reasons:

- When fast path is enabled, TCP checksum computation is automatically offloaded to the network interfaces.

  Note: Only specific NICs support this functionality.
- Route lookup to the final destination is skipped when fast path is enabled.

Differences between fast path over IPv4 and IPv6

Fast path over IPv4 provides load balancing between multiple network interfaces on the same subnet because responses are sent on the same network interface that receives the incoming requests. IPv4 uses the same source IPv4 address and the source MAC address of the incoming packet in the destination packet.

Fast path over IPv6 does not provide load balancing because it uses the default gateway of the incoming interface as the destination. Fast path over IPv6 always performs an NDP lookup to find the MAC address of the next hop. Therefore, the responses might not be sent on the same interface that receives the request.

How to manage the routing table

You can manage the routing table automatically by using the `routed` daemon, or manually by using the `route` command.

What the routed daemon does

The `routed` daemon performs several functions automatically, such as deleting redirected routes, and can be configured to perform several additional functions, such as controlling RIP and IRDP behavior. The `routed` daemon is enabled by default.

The `routed` daemon performs the following functions by default:

- Deletes redirected routes after a specified period
- Performs router discovery with ICMP Router Discovery Protocol (IRDP)
  Router discovery is useful only if there is no static default route.
- Listens for Routing Information Protocol (RIP) packets
- Migrates routes to alternate interfaces when multiple interfaces are available on the same subnet
  For example, the network interface e0a is on 10.1.1.2/24 and e0b is on 10.1.1.3/24. The default route to 10.1.1.1 is through e0a. The subnet route 10.1.1.0/24 is also through e0a. Now if e0a is brought to the down status, the `routed` daemon moves the default route to e0b and the subnet route to e0b. Therefore, this function ensures continued connectivity to the local subnet and remote networks.

The `routed` daemon can also be configured to perform the following functions:

- Control RIP and IRDP behavior
- Generate RIP response messages that update a host route on your storage system
- Recognize distant gateways identified in the `/etc/gateways` file
- Authenticate RIPv2 advertisements with Message Digest 5 (MD5) algorithm for improved network security

Note: The `routed` daemon supports only IPv4.

For more information about the `routed` daemon, see the na_routed(1) man page.

When the routed daemon should be turned off

In some circumstances, you should turn off the `routed` daemon. For example, if you have multiple interfaces on the same subnet and you want to direct network traffic to a specific interface, turn off the `routed` daemon and use a static configuration.

You can turn off the `routed` daemon if the following conditions are true:

- You do not use RIP or router discovery.
- You have a single router per subnet or a network in which redirects are not sent.
- You can manage your routing table directly.

Note: If you use dynamic routing, you are advised to keep the `routed` daemon `on` because turning the `routed` daemon `off` might cause unexpected routing behavior. If you are using static routing, make sure the entries are correct by using the `netstat -r` command to check the static routing configuration.

Related tasks

Enabling or disabling the routed daemon from the command-line interface on page 58

How dynamic routing works for IPv6

IPv6 routing table entries are created by default when you enable IPv6. Additional entries are added dynamically in the routing table on receiving Router Advertisement and ICMP redirect messages.

Storage systems populate a default router list and a prefix list, based on the information in the Router Advertisement messages. The default router list is used to select a router for off-link destinations, and the prefix list is used to determine whether a destination address is on-link.

Related tasks

Enabling or disabling IPv6 on page 47

Routing tables in a vFiler unit environment

If you enable the MultiStore license and have more than one IPspace, Data ONTAP disables the `routed` daemon. Therefore, routing tables in such vFiler unit environments must be managed manually with the `route` command.

The `routed` daemon runs only in the default IPspace. Data ONTAP does not support running multiple instances of the `routed` daemon in multiple IPspaces. Therefore, if you have multiple IPspaces, the `routed` daemon is automatically disabled.

All vFiler units in an IPspace share a routing table. Therefore, any commands that display or manipulate the routing table apply to all vFiler units in that IPspace.

Circumstances that might alter the routing table

Certain events can cause the routing table to be modified. You should check the routing table after these events occur to be sure that it is still configured as required.

The routing table might be modified in the following circumstances:

- A new interface is configured with the `ifconfig` command and there are no existing entries for the new network number in the routing table.
- You use the `route add` command to add an entry to the routing table.
- Your storage system receives an ICMP/ICMPv6 redirect packet, which notifies the storage system of a better first-hop router for a particular destination.

  Note: Your storage system ignores ICMP/ICMPv6 redirect packets if the `ip.icmp_ignore_redirect.enable` option is turned on.
- Your storage system is rebooted after the default route in the `/etc/rc` file is modified.
- The default route is added to the routing table on receiving an IPv6 Router Advertisement message.

Specifying the default route

The routing table uses the default route when no other known route exists for a given packet's destination address. You can enter the IP address or the name for the default route either during initial setup process of Data ONTAP. You can also set the default route, later, by modifying the `/etc/rc` file.

Before you begin

The dedicated management interface e0M should not be configured with any of the IP addresses belonging to the same subnet as that of the default route gateway.

An EMS message is displayed if the default route is configured on e0M.

Step

1. Add the following command to the `/etc/rc` file:

   `route add inet6 default` ***`gateway metric`***

   `gateway` is the IP address of the default route.

   `metric` is the value given to each manually configured route.

   Example
   You can set the default route in the `/etc/rc` file by using the `route add` command:

```
hostname sys1
ifconfig e0 192.0.2.21 netmask 255.255.255.0 mediatype 100tx-fd
route add default 192.0.2.1 1
routed on
```

Related concepts

Guidelines for configuring e0M on page 14

How to enable or disable routing mechanisms

Both the fast path mechanism and the `routed` daemon are enabled by default in Data ONTAP. You can enable or disable these routing mechanisms using the command-line interface.

If you disable both fast path and the `routed` daemon, you must configure routing manually.

Enabling or disabling fast path

IP fast path is a mechanism that uses the network interface of an inbound request to send the response by bypassing a routing table lookup. Fast path is enabled by default for all TCP and NFS UDP (NFS/UDP) connections.

About this task

- In TCP connections that use an asymmetric network path, fast path is disabled automatically on the third retransmission of a data packet.
  All the subsequent packets follow the route specified in the routing table.
- Disabling fast path disables TCP segmentation offload (TSO).
- During a takeover, the routing behavior on the failed node is determined by the fast path settings on the surviving node.
- When fast path is enabled, TCP checksum computation is automatically offloaded to the network interfaces.

Step

1. To enable or disable fast path, enter the following command:

   `options ip.fastpath.enable {on|off}`

   `on`—Enables fast path

   `off`—Disables fast path

   Note: You can use the `netstat -x` command to check if fast path is enabled.

Related concepts

How fast path works on page 53

Enabling or disabling the routed daemon from the command-line interface

You can manage the routing table automatically by using the routed daemon. You can turn on or turn off the routed daemon by using the `routed` command.

About this task

You must add the `routed` command to the `/etc/rc` file for the routed daemon behavior to persist across storage system reboots.

Step

1. To enable or disable the routed daemon, enter the following command:

   `routed {on|off}`

`on`—Turns on the routed daemon

`off`—Turns off the routed daemon

Related concepts

What the routed daemon does on page 55
When the routed daemon should be turned off on page 55

Related references

The routed daemon on page 184

How to view the routing table and default route information

You can view the routing table of the storage system and default route information relating to your route's destinations, their gateways, how much each route is used, and the interface used by each route. Flags showing route status information are also displayed.

Viewing the routing table from the command-line interface

You can view information such as default route and the routes for specific destination addresses. If you have enabled the IPv6 option, the routing table displays both the IPv4 and IPv6 information.

Step

1. To view the Data ONTAP routing table, enter one of the following commands:
   - **`netstat -rn`**
   - **`route -s`**

Example for interpreting the routing table

The output of the `netstat -rn` command is as follows:

```
netstat -rn
Routing tables

Internet:
Destination
Gateway                                    Flags
Refs                Use     Interface
default
192.0.2.1                  UGS
3                    21397    e0a
127.0.0.1
127.0.0.1
UH                                 0
0            lo
192.0.2/24                link#11
UC                                 0
```

```
0          e0a
192.0.2.1                    0:d0:d3:0:30:0
UHL                             1
0          e0a
192.0.2.23              0:1:30:b8:30:c0
UHL                             0
0             e0a
192.0.2.24              0:1:30:b8:2e:c0
UHL                             0
0             e0a

Internet v6:
Destination      Gateway            Flags     Refs     Use  Interface
default          fe80::21b:2bff:fed7:ec00%e1a UG 0        0  e1a
::1              ::1                UH          0        0  lo
2001:0db8::/64   link#3             UC          0        0  e1a
2001:0db8:b255:4213::/64 link#3                    UC
0        0  e1a
2001:0db8:b255:4213::1 link#3                             UHL
0        0  e1a
```

In this example, the destination can be a host 192.0.2.1, a network 192.0.2/24, or the default route. If the destination is a subnet on a network, the network number is followed by a forward slash (/) and a number that describes the network mask for that network.

The IPv6 routing table also has the same network parameters except that the network mask is replaced by the prefix length for that network.

Routing table flags

The following table describes the Flags column in the `netstat -rn` output.

Flag	Description
U	Up—Route is valid
G	Gateway—Route is to a gateway router rather than to a directly connected network or host
H	Host name—Route is to a host rather than to a network, where the destination address is a complete address
R	Reject—Set by ARP when an entry expires (for example, the IP address could not be resolved into a MAC address)
D	Dynamic—Route added by a route redirect or RIP (if routed is enabled)
M	Modified—Route modified by a route redirect
C	Cloning—A new route is cloned from this entry when it is used
L	Link—Link-level information, such as the Ethernet MAC address, is present
S	Static—Route added with the `route` command

Viewing the default route information from the command-line interface

You can view default route information such as whether the `routed` daemon is turned on or off, default route information, and routing protocols. You can view the default route information by using the `routed status` command.

Step

1. Enter the following command:

   `routed status`

   Note: You can also view the default route by using the `netstat -rn` or `route -s` commands.

   Example

   The output of the `routed status` command is as follows:

```
routed status
RIP snooping is on
Gateway                 Metric  State   Time Last Heard
example-gateway.com       1     ALIVE   Wed Mar 18 13:58:56 IST 2009
0 free gateway entries, 1 used
```

   In the `routed status` command output, metric is the route property that is used to determine the preferred route. The route with the lowest metric is the preferred route. You should always use a metric greater than 0 when adding default routes.

Modifying the routing table

You might want to add or delete routes in your routing table depending on the changes in your network. You can use the `route` command to modify the routing table.

Step

1. Depending on whether you want to add or delete a route from the routing table, perform the following step:

If you want to...	Enter the following command...
Add a route	**route add *destination* [*gateway metric*]** *destination* is the IP address or host name of the destination for which the route is being added or deleted. *gateway* is the gateway for the specified *destination*. *metric* indicates the number of hops to the *destination*. The value of *metric* should be greater than zero when the route to the destination is through the *gateway*. The value of *metric* is zero when the *destination* is on a directly-attached network.
Delete a route	**route delete *destination* [*gateway metric*]** **Attention:** You must not delete a cloned route (denoted by the C flag) from the routing table; if you do, the network connectivity to that subnet is lost. If you have deleted a cloned route, you must add the route again to the routing table in either of the following ways: • Bring the interface that connects to the particular subnet first to the down state and then to the up state. You can change the state of the interface by using the ifconfig command. • Delete and reconfigure the IP address on the interface that connects to the particular subnet.

For more information about the route command and options, see the na_route(1) man page.

Example

To add a destination with the IP address 192.0.2.25 to the routing table, enter the following command:

```
route add 192.0.2.25 gateway.com 1
```

You can verify that the route to this destination is added to the routing table by using the netstat -rn or route -sn command, as shown in the following output:

```
system1> netstat -rn
Routing tables

Internet:
Destination      Gateway            Flags     Refs     Use  Interface
default          192.0.2.1          UGS          4   184855  e0a
127.0.0.1        127.0.0.1          UH           0        0  lo
192.0.2/24       link#11            UC           2     1238  e0a
192.0.2.1        0:d0:d3:0:30:0     UHL          0       40  e0a
192.0.2.23       0:1:30:b8:30:c0    UHL          1        0  e0a
192.0.2.25       192.0.2.1          UHL          0     1285  lo
```

In this example, the subnet route, 192.0.2, is a cloned route.

Related tasks

Changing the status of an interface on page 34

Related references

Routing table flags on page 60

How to maintain host-name information

Data ONTAP relies on correct resolution of host names to provide basic connectivity for storage systems on the network. If you are unable to access the storage system data or establish sessions, there might be problems with host-name resolution on your storage system or on a name server.

Host-name information can be maintained in one or all of the following ways in Data ONTAP:

- In the `/etc/hosts` file on your storage system's default volume
- On a Domain Name System (DNS) server
- On a Network Information Service (NIS) server

If you use more than one of the resources for host-name resolution, the order in which they are used is determined by the `/etc/nsswitch.conf` file.

How the /etc/hosts file works

Data ONTAP uses the `/etc/hosts` file to resolve host names to IP addresses. You need to keep the `/etc/hosts` file up-to-date. Changes to the `/etc/hosts` file take effect immediately.

When Data ONTAP is first installed, the `/etc/hosts` file is automatically created with default entries for the following interfaces:

- Local host
- All interfaces on your storage system

The `/etc/hosts` file resolves the host names for the storage system on which it is configured. This file cannot be used by other systems for name resolution.

For more information about file formats, see the na_hosts(5) man page.

You can add IP address and host name entries in the `/etc/hosts` file in the following two ways:

- Locally—You can add entries by using the command-line interface.
- Remotely—If the file has many entries and you have access to an NIS makefile master, you can use the NIS makefile master to create the `/etc/hosts` file. This method prevents errors that might be caused by editing the file manually.

Adding a host name in the /etc/hosts file

You can add the host name and aliases of the storage system in the `/etc/hosts` file. You can use the `setup` command to rewrite the `/etc/hosts` file.

About this task

During setup, if you enable IPv6 on the storage system and configure IPv6 addresses for your network interfaces, these IPv6 addresses are also added to the `/etc/hosts` file.

Step

1. From a workstation that has access to your storage system, edit the `/etc/hosts` file. Add the following line to the `/etc/hosts` file:

   IP_address host_name aliases

   IP_address is the IP address of the host.

   host_name is the name of the host.

   aliases are the alias names for the host.

   #### Example

   To add a host name, myhost, with an IP address 192.0.2.16, add the following line in the `/etc/hosts` file:

   192.0.2.16 myhost newhost myhost-e0a

   newhost and myhost-e0a are the alias names for myhost.

   The following is a sample `/etc/hosts` file:

```
#Auto-generated by setup Tue Apr 21 17:41:40 IST 2009
127.0.0.1 localhost
192.0.2.16         myhost myhost-e0a
# 0.0.0.0       myhost-e0b
# 0.0.0.0       myhost-e0c
# 0.0.0.0       myhost-e0d
```

   The following is a sample `/etc/hosts` file in which an IPv6 address is also configured for the interface e0a:

```
#Auto-generated by setup Tue Apr 21 17:41:40 IST 2009
127.0.0.1 localhost
192.0.2.16         myhost myhost-e0a
2001:0db8::95   myhost myhost-e0a
# 0.0.0.0       myhost-e0b
# 0.0.0.0       myhost-e0c
# 0.0.0.0       myhost-e0d
```

Hard limits for the /etc/hosts file

You need to be aware of the hard limits on the line size and number of aliases when you edit the `/etc/hosts` file.

The hard limits are as follows:

- Maximum line size is 1022 characters.
  The line size limit includes the end of line character. You can enter up to 1021 characters per line.
- Maximum number of aliases is 34.

Note: There is no limit on file size.

Changing the host name of a storage system

You can change the host name of a storage system by editing the `/etc/hosts` file, and then using the `hostname` command.

Steps

1. Edit the `/etc/hosts` file to include the new host name of the storage system.
2. At the command-line interface of the storage system, enter the following command to specify a new name for the host:

   `hostname` ***`new_name`***

   `new_name` is the new host name of the storage system. The host name must match the entry made in the `/etc/hosts` file in Step 1.

   Attention: If you skip this step, any manual or scheduled SnapMirror operations might fail. Use the `hostname` command to specify the correct name before any SnapMirror operations are initiated.

How to configure DNS to maintain host information

You can maintain host information centrally using DNS. With DNS, you do not have to update the `/etc/hosts` file every time you add a new host to the network.

If you have several storage systems on your network, maintaining host information centrally saves you from updating the `/etc/hosts` file on each storage system every time you add or delete a host.

If you configure DNS later, you must take the following actions:

- Specify DNS name servers.
- Specify the DNS domain name of your storage system.
- Enable DNS on your storage system.

If you want to use primarily DNS for host-name resolution, you should specify it ahead of other methods in the hosts section of the `/etc/nsswitch.conf` file.

Correct host-name resolution depends on correctly configuring the DNS server. If you experience problems with host-name resolution or data availability, check the DNS server in addition to local networking.

Related concepts

How the /etc/hosts file works on page 64

Configuring DNS from the command-line interface

You can configure your storage system to use one or more DNS servers for host-name resolution. You can configure DNS by first creating or editing the `/etc/resolv.conf` file, then specifying the DNS domain name, and finally enabling DNS through the command-line interface.

Steps

1. Depending on whether you want to create or edit the `/etc/resolv.conf` file, perform the following step:

If you are...	Then...
Creating the `/etc/resolv.conf` file	By using a text editor, create the `/etc/resolv.conf` file in the root volume. The file can consist of up to three lines, each specifying a name server host in the following format. **`nameserver ip_address`** *`ip_address`* is the IP address of the DNS name server. The IP address can be an IPv4 or an IPv6 address. **Note:** If an IPv6 link-local address is specified as a DNS name server, the address must be appended with `%`*`interface_name`*. The appended *`interface_name`* is the name of the interface on the storage system that is connected to the same link as the specified DNS name server. For example: **`nameserver 2001:0db8::85a3:0:0:8a2e:0370:99`** e0a is the interface on the storage system that is connected to the same link as the DNS name server with the IPv6 address 2001:0db8::85a3:0:0:8a2e:0370:99.
Editing the `/etc/resolv.conf` file	From a workstation that has access to the root volume of your storage system, edit the `/etc/resolv.conf` file using a text editor.

2. Enter the following command to specify the DNS domain name:

   `options dns.domainname` ***`domain`***

   `domain` is the new domain name, which follows the host name of your storage system in the fully qualified domain name.

3. Enter the following command to enable DNS:

   options dns.enable {on|off}

   on—Enables DNS

   off—Disables DNS

Hard limits for the /etc/resolv.conf file

You need to be aware of the hard limits for name servers, domain name, and search domains when you create or edit the /etc/resolv.conf file.

The hard limits for the /etc/resolv.conf file are as follows:

- Maximum line size is 256.
- Maximum number of name servers is 3.
- Maximum domain name length is 256 characters.
- Maximum search domains limit is 6.

  Note: You should use only tab or space to separate host names in the search domain list.

- Total number of characters for all search domains is 256.

Note: There is no limit on file size.

How DNS resolves host names

DNS uses certain records for resolving a domain name to an IP address. To determine a host name based on the IP address, DNS uses the reverse lookup.

For resolving IPv4 addresses, DNS uses the A record. The A record can store a 32-bit address and can resolve IPv4 addresses. To resolve IPv6 addresses, DNS uses the AAAA record. The AAAA record can store a 128-bit address and can resolve IPv6 addresses.

IPv4 reverse DNS lookups use the in-addr.arpa domain. An IPv4 address is represented in the in-addr.arpa domain by a sequence of bytes, represented as decimal numbers, in reverse order. The numbers are separated by dots and end with the suffix .in-addr.arpa.

IPv6 reverse DNS lookups use the ip6.arpa domain. An IPv6 address is represented as a name in the ip6.arpa domain by a sequence of nibbles, represented as hexadecimal digits, in reverse order. These nibbles are separated by dots and end with the suffix .ip6.arpa.

The following table shows sample IPv4 and IPv6 addresses and their reverse DNS lookups:

IP address	Reverse lookup domain name
192.0.2.10	10.2.0.192.in-addr.arpa
2001:0db8:85a3:0:0:8a2e:0370:99	9.9.0.0.0.7.3.0.e.2.a.8.0.0.0.0.0.0.0.0.3.a.5.8.8.b.d.0.1.0.0.2.ip6.arpa

DNS name caching

DNS name caching speeds up the process whereby the DNS name resolver converts host names into IP addresses. The DNS name cache stores DNS requests so that they can be easily and quickly found when needed. DNS name caching is enabled by default.

Name caching improves DNS performance during a name server failover and reduces the time needed for an HA pair takeover and giveback.

You can disable DNS name caching by using the `dns.cache.enable` option, but doing so might have an adverse performance impact. The `dns flush` command removes all entries from the DNS name cache. However, the command has no effect if DNS name caching is not enabled.

For more information about the `dns flush` command and the `dns.cache.enable` option, see the na_ dns(1) man page.

DNS information you can view

You can view information about whether DNS and DNS name caching are enabled, configured name servers, state of these name servers (whether up or down), configured DNS domain name, DNS name cache statistics, and performance statistics for each name server.

The `dns info` command displays the status of the DNS resolver. If DNS is enabled, the command displays the following information:

- Whether DNS is enabled
- Whether DNS name caching is enabled
- Caching statistics
  - Cache hits: Number of DNS requests that were found in the cache
  - Cache misses: Number of DNS requests that were not found in the cache and that required a DNS query to the name server
  - Cache entries: Number of entries currently in the DNS name cache
  - Expired cache entries
  - Number of cache replacements
- Details about each name server that was polled by your storage system:
  - IP address of the DNS server
  - State of the name server, displayed as "UP," "DOWN," or "NO INFO"
  - Date of the last DNS request to that name server
  - Average time in milliseconds for a DNS query
  - Number of DNS queries made
  - Number of DNS queries that resulted in errors
- Default DNS domain name of the storage system
- Search domains of the storage system

The search domains are domain suffixes that are used to convert unqualified domain names into fully qualified domain names (FQDN). The search domains are read from the `/etc/resolv.conf` file.

For more information about the `dns info` command and the resulting display, see the na_ dns(1) man page.

How to use dynamic DNS to update host information

You can use dynamic DNS updates to prevent errors and save time when sending new or changed DNS information to the primary master DNS server for your storage system's zone. Dynamic DNS allows your storage system to automatically send information to the DNS servers as soon as the information changes on the system.

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.

For example, if you want to change the IP address on interface e0 of *`storagesystem1`*, you can simply configure e0 with the new IP address. The storage system *`storagesystem1`* automatically sends its updated information to the primary master DNS server.

Note: Data ONTAP supports a maximum of 64 Dynamic Domain Name Server (DDNS) aliases.

How dynamic DNS updates work in Data ONTAP

If dynamic DNS updates are enabled on your storage system, Data ONTAP periodically sends updates to the primary master DNS server for its zone. Updates are also sent if any DNS information changes on your system.

Your storage system finds the primary master DNS server for its zone by querying the DNS servers configured in your storage system's `/etc/resolv.conf` file. The primary master DNS server might be different from the ones configured in your storage system's `/etc/resolv.conf` file.

By default, periodic updates are sent every 12 hours. A time-to-live (TTL) value is assigned to every DNS update sent from your storage system. The TTL value defines the time for which a DNS entry is valid on the DNS server. By default, the TTL value is set to 24 hours, and you can change it.

When your storage system sends an update to the DNS server, it waits up to five minutes to receive an acknowledgement of the update from the server. If it does not receive an acknowledgement, the storage system sends the update again. This time, the storage system doubles the waiting interval (to 10 minutes), before sending the update. The storage system continues to double the waiting interval with each retry until a waiting interval of 160 minutes or TTL/2, whichever is less, is reached.

Support for dynamic DNS updates in Data ONTAP

When you use dynamic DNS updates in Data ONTAP, you must be aware of certain conditions, such as the types of systems and network interfaces that support dynamic DNS, secure updates, and behavior of vFiler units with dynamic DNS.

The following conditions apply to dynamic DNS updates:

- By default, dynamic DNS updates are disabled in Data ONTAP.
- Dynamic DNS updates are supported on UNIX and Windows systems.
- On Windows DNS servers, secure dynamic DNS updates can be used to prevent malicious updates on the DNS servers. Kerberos is used to authenticate updates.

  Even if secure dynamic DNS updates are enabled, your storage system initially tries sending updates in clear text. If the DNS server is configured to accept only secure updates, the updates sent in clear text are rejected. Upon rejection, the storage system sends secure DNS updates.
- For secure dynamic DNS updates, your storage system must have CIFS running and must be using Windows Domain authentication.
- Dynamic DNS updates can be sent for the following:
  - Physical interfaces
  - Interface group and VLAN interfaces
  - vFiler units
- You cannot set TTL values for individual vFiler units. All vFiler units inherit the TTL value that is set for vfiler0, which is the default vFiler unit and is the same as the physical storage system.
- DHCP addresses cannot be dynamically updated.
- In a takeover situation, the hosting storage system is responsible for sending DNS updates for IP addresses for which it is responding.
- For both manual and autoconfigured global IPv6 unicast addresses, the dynamic DNS update is sent after Duplicate Address Detection is performed. For IPv6 addresses of any other type and scope, your storage system does not send any dynamic DNS update.

Enabling or disabling dynamic DNS updates

Dynamic DNS enables your storage system to automatically send information to the DNS servers as soon as the information changes on the system. By default, dynamic DNS is disabled on the storage system. You can enable dynamic DNS on your storage system by using the `options dns.update.enable` command.

Step

1. Enter the following command:

   `options dns.update.enable {on|off|secure}`

   `on`—Enables dynamic DNS updates

   `off`—Disables dynamic DNS updates

`secure`—Enables secure dynamic DNS updates

Note: Secure dynamic DNS updates are supported for Windows DNS servers only.

Disabling the transmission of DNS updates for an IP address

You can disable the transmission of dynamic DNS updates for an IP address by using the `ifconfig` command. You can also disable the transmission of dynamic DNS updates for an IP address that is configured on e0M.

About this task

- You should not disable dynamic DNS updates for an interface that is part of an interface group.
- You should disable dynamic DNS updates for an IP address that is configured on dedicated management interface (e0M).
- You can also disable dynamic DNS updates for an IPv6 address.

Step

1. Enter the following command:

   ifconfig *interface_name* no_ddns *IP_address*

   interface_name is the name of the interface.

   IP_address is the IP address of the interface.

   Example
   Use the following command to ensure that dynamic DNS updates are not sent from the interface e0a:

   ifconfig e0a no_ddns 192.0.2.30

   The following output shows the output of the `ifconfig` command after the dynamic DNS is disabled for the interface:

```
ifconfig e0a
e0a: flags=0x2d48867<UP,BROADCAST,RUNNING,MULTICAST,TCPCKSUM> mtu 1500
inet 192.0.2.30 netmask 0xff000000 broadcast 10.255.255.255 noddns
ether 00:a0:98:07:66:02 (auto-1000t-fd-up) flowcontrol full
```

   The `ifconfig` command output shows the "noddns" keyword that indicates that dynamic DNS updates are disabled for this IP address.

Changing the time-to-live setting for DNS entries

You can change the time-to-live setting for DNS entries by using the `options dns.update.ttl` command.

Step

1. Enter the following command:

   `options dns.update.ttl` ***`time`***

   `time` can be set in seconds (**`s`**), minutes (**`m`**), or hours (**`h`**), with a minimum value of 600 seconds and a maximum value of 24 hours.

   Example
   To set the TTL to two hours, enter the following command:

   `options dns.update.ttl 2h`

Related concepts

How dynamic DNS updates work in Data ONTAP on page 70

How to use NIS to maintain host information

NIS enables you to centrally maintain host information. In addition, NIS enables you to maintain user information.

NIS provides the following methods for resolving the storage system's host name:

- Using the `/etc/hosts` file on the NIS server
  You can download the `/etc/hosts` file on the NIS server to your storage system's default volume for local host-name lookup.
- Using a hosts map that is maintained as a database on the NIS server
  The storage system uses the hosts map to query during a host lookup request across the network.
- Using the ipnodes map that is maintained as a database on the NIS server
  The ipnodes map is used for host lookup when IPv6 is enabled on your storage system.

  Note: The ipnodes database is supported only on Solaris NIS servers. To resolve a host name to an address, your storage system (with IPv6 enabled) first looks in the ipnodes database. If the IP address is not present in the ipnodes database, the application looks in the hosts database. However, if IPv6 is not enabled, then your storage system looks only in the hosts database and does not refer the ipnodes database.

How using NIS slaves can improve performance

Host-name resolution by using a hosts map can have a performance impact because each query for the hosts map is sent across the network to the NIS server. You can improve the performance of your storage system by downloading the maps and listening for updates from the NIS master server.

The NIS slave improves performance by establishing contact with an NIS master server and performing the following two tasks:

- Downloading the maps from the NIS master server
  You can download the maps from the NIS master server to the NIS slave by running the `yppush` command from the NIS server. You can also download the maps by disabling and then enabling the NIS slave from your storage system. After the maps are downloaded, they are stored in the `/etc/yp/nis_domain_name` directory. The NIS slave then services all the NIS requests from your storage system by using these maps. The NIS slave checks the NIS master every 45 minutes for any changes to the maps. If there are changes, they are downloaded.
- Listening for updates from the NIS master
  When the maps on the NIS master are changed, the NIS master administrator can optionally notify all slaves. Therefore, in addition to periodically checking for updates from the NIS master, the NIS slave also listens for updates from the master.

You cannot configure the NIS slave during the setup procedure. To configure the NIS slave after the setup procedure is complete, you need to enable NIS slave by setting `options nis.slave.enable` to `on`.

Note: The NIS slave does not respond to remote NIS client requests and therefore cannot be used by other NIS clients for name lookups.

Related concepts

Guidelines for using NIS slaves on page 75

Related tasks

Enabling an NIS slave on your storage system on page 79

How an NIS master is selected

If you enable the NIS slave on your storage system, the NIS servers listed with the `nis.servers` option are contacted to determine the master NIS server.

The NIS master can be different from the servers that are listed with the `nis.servers` option. In such a case, the servers listed with the `nis.servers` option inform the slave about the master server.

The NIS slave on your storage system can contact the master only if any one of the following conditions is true:

- The NIS server has an entry in the ipnodes map for the master.

- The NIS server has an entry in the hosts map for the master.
- The `/etc/hosts` file on your storage system is able to resolve the IP address of the master.

Creating /etc/hosts from the NIS master

You can create a host file remotely and modify the NIS master to install the host file in the `/etc` directory. This method is useful if you have many entries in your host file.

Steps

1. On the NIS server, open the NIS Makefile with a text editor.
2. Locate the section for `hosts.time`.
3. Add the following lines at the end of the `hosts.time` section, replacing *`dirname`* with a directory name of your choice, and *`toaster 1`*, *`toaster2`*, and so on with names of the storage systems:

   ```
   @mntdir=/tmp/dirname_etc_mnt_$$$$;\ if [ ! -d $$mntdir ]; then rm -f $
   $mntdir; \ mkdir $$mntdir; fi;\ for s_system in toaster1 toaster2
   toaster3 ; do \ mount $$s_system:/etc $$mntdir;\ mv $$mntdir/hosts $
   $mntdir/hosts.bak;\ cp /etc/hosts $$mntdir/hosts;\ umount $$mntdir;\
   done;\ rmdir $$mntdir
   ```

4. Save the NIS Makefile.

   The `/etc/hosts` file on your storage system is updated whenever the NIS Makefile is run.

Related concepts

How the /etc/hosts file works on page 64

Guidelines for using NIS slaves

When using an NIS slave, you should follow certain guidelines, such as the available space in the storage system, conditions for enabling DNS, and supported configurations.

The following guidelines apply when using the NIS slave:

- The root volume of your storage system must have sufficient space to download maps for the NIS slave. Typically, the space required in the root volume is same as the size of the maps on the NIS server.
- If the root volume does not have enough space to download maps, the following occurs:
  - An error message is displayed informing you that the space on the disk is not sufficient to download or update the maps from the NIS master.
  - If the maps cannot be downloaded, the NIS slave is disabled. Your storage system switches to using hosts map on the NIS server for name resolution.
  - If the maps cannot be updated, your storage system continues to use the old maps.
- If the NIS master server was started with the `-d` option or if the `hosts.byname` and `hosts.byaddr` maps are generated with the `-b` option, your storage system must have DNS

enabled, DNS servers must be configured, and the hosts entry in the `/etc/nsswitch.conf` file must contain DNS as an option to use for host name lookup.

If you have your NIS server configured to perform host name lookups using DNS, or if you use DNS to resolve names that cannot be first resolved using the `hosts.by*` maps, using the NIS slave causes those lookups to fail. This is because when the NIS slave is used, all lookups are performed locally using the downloaded maps. However, if you configure DNS on your storage system, the lookups succeed.

You can use the NIS slave for the following:

- Interface groups and VLAN interfaces
- vFiler units
- HA pairs

Note: In an HA pair, you should ensure that the `nis.servers` options value is the same on both nodes and that the `/etc/hosts` file on both nodes can resolve the name of the NIS master server.

Related concepts

How using NIS slaves can improve performance on page 74

NIS administrative commands

You can use the NIS administrative commands to view the NIS server information.

Data ONTAP supports the standard NIS administrative commands listed in the following table. For more information, see each command's man page.

Command	Function
`ypcat`	Prints an entire NIS map.
`ypgroup`	Displays the NIS group cache entries.
`ypmatch`	Looks up specific entries in an NIS map.
`ypwhich`	Returns the name of the current NIS server.

How to configure NIS with Data ONTAP interfaces

You can configure your storage system to use one or more NIS servers either during the setup procedure or later using the Data ONTAP command-line interface.

If you want to use NIS primarily for host-name resolution, specify it ahead of other methods in the hosts map in the `/etc/nsswitch.conf` file.

To configure NIS, you need to do all of the following:

- Specify the NIS server to which your storage system should bind.
- Specify the NIS domain name of your storage system.
- Enable NIS on your storage system.

Correct host-name resolution depends on correctly configuring the NIS server. If you experience problems with host-name resolution or data availability, check the NIS server in addition to local networking.

For more information about your NIS client, see the na_nis(1) and na_nis(8) man pages.

Enabling NIS using the command-line interface

You can enable NIS on your storage system for host-name resolution. Whenever you enable NIS for group lookup services, you must also enable local caching of the NIS group files.

About this task

If you use NIS for group lookup services, disabling NIS group caching can severely degrade performance. Failure to enable NIS lookups and NIS caching together could lead to timeouts as CIFS clients attempt authentication.

Steps

1. To enable NIS lookups, enter the following command:

   `options nis.enable on`

2. To enable NIS caching, enter the following command:

   `options nis.group_update.enable on`

Specifying the NIS domain name

You can specify the NIS domain name to which your storage system belongs.

Step

1. Enter the following command:

   `options nis.domainname` ***`domain`***

   `domain` is the NIS domain name to which your storage system belongs. For example, typical NIS domain names might be `sales` or `marketing`. The NIS domain name is usually not the same as the DNS domain name.

Specifying NIS servers to bind to your storage system

You can specify an ordered list of NIS servers to which you want your storage system to bind. The list should begin with the closest NIS server (closest in network terms) and end with the farthest one.

About this task

Keep the following in mind before performing the binding procedure:

- Using the NIS broadcast feature can incur security risks.
- You can specify NIS servers by IP address or host name. If host names are used, ensure that each host name and its IP address are listed in the `/etc/hosts` file of your storage system. Otherwise, the binding with the host name fails.
- You can only specify IPv4 addresses or server names that resolve to IPv4 addresses by using the `/etc/hosts` file on your storage system.

Step

1. Enter the following command to specify the NIS servers and their order:

   options nis.servers *ip_address, server_name,*[*]

   The asterisk (*) specifies that broadcast (for IPv4) and multicast (for IPv6) is used to bind to NIS servers if the servers in the list are not responding. The '*' is the default value. If you do not specify the broadcast or multicast option, and none of the listed servers is responding, NIS services are disrupted until one of the preferred servers responds.

Example

The following command lists two servers and uses the default broadcast (multicast for IPv6) option:

```
options nis.servers 192.0.2.1,nisserver-1,*
```

Your storage system first tries to bind to 192.0.2.1. If the binding fails, the storage system tries to bind to nisserver-1. If this binding also fails, the storage system binds to any server that responds to the broadcast or multicast. However, the storage system continues to poll the preferred servers. When one of the preferred server is found, the storage system binds to the preferred server.

The following command lists an NIS server with an IPv6 address and uses the default multicast option:

```
options nis.servers 2001:0db8:85a3:0:0:8a2e:0370:99,*
```

Related concepts

How an NIS master is selected on page 74

Enabling an NIS slave on your storage system

You can enable an NIS slave on your storage system to reduce traffic over your network.

About this task

If you enable IPv6 on your storage system, your storage system can have multiple addresses configured for it in the host-name database. These addresses appear in the host-name lookup, depending on the following conditions:

- If you disable the NIS slave, you can obtain all the addresses from either the hosts database or the ipnodes database in the NIS server.
- If you disable the NIS slave, your storage system reverts to the original configuration, in which it contacts an NIS server to resolve host names.
- If you enable the NIS slave, only the last address from the list of addresses available in the `/etc/hosts` file is stored for a host name in the host database downloaded to your system.
- If you enable the NIS slave, a maximum of three addresses are stored for a host name in the ipnodes database downloaded to your system. At least one address from each address family is stored.

Step

1. To enable or disable an NIS slave on your storage system, enter the following command:

   `options nis.slave.enable {on|off}`

Related concepts

How using NIS slaves can improve performance on page 74
Guidelines for using NIS slaves on page 75

What NIS information you can view

You can view information about NIS master and slave servers, netgroup caches, and performance statistics.

The `nis info` command displays the following types of NIS information:

- NIS domain name
- Last time the local group cache was updated
- Information about each NIS server that was polled by your storage system:
  - IP address of the NIS server
  - Type of NIS server
  - State of the NIS server
  - Whether your storage system is bound to the NIS server

    - Time of polling
- Information about the NIS netgroup cache:
    - Status of the cache
    - Status of the "*.*" entry in the cache
    - Status of the "*.nisdomain" entry in the cache
- Whether an NIS slave is enabled
- NIS master server
- Last time the NIS map was checked by the NIS slave
- NIS performance statistics:
    - Number of YP lookup network retransmission
    - Total time spent in YP lookups
    - Number of network retransmission
    - Minimum time spent in a YP lookup
    - Maximum time spent in a YP lookup
    - Average time spent in a YP lookup
- Response statistics for the three most recent YP lookups

For more information about the `nis info` command and resulting display, see the na_nis(1) man page.

Viewing NIS performance statistics

You can use the `nis info` command to view NIS performance statistics for your storage system.

Step

1. Enter the following command to view NIS information:

   `nis info`

Example

The following example shows the statistics provided by the `nis info` command.

```
system1*> nis info
NIS domain is lab.example.com
    NIS group cache has been disabled

     IP Address    Type State Bound   Last Polled
Client
calls Became Active
-------------------------------------------------------------------------
-----------------------------
    192.0.2.12 PREF ALIVE YES     Mon Jan 23 23:11:14
GMT 2008 0 Fri Jan 20 22:25:47 GMT 2008

                  NIS Performance Statistics:
               Number of YP Lookups: 153
```

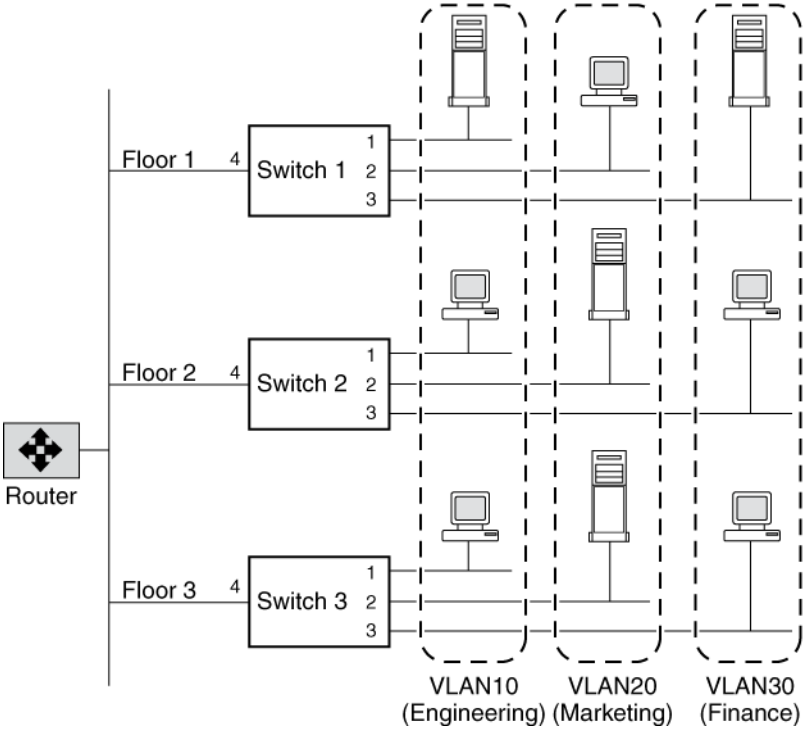
```
               Total time spent in YP Lookups: 684 ms, 656 us
               Number of network re-transmissions: 0
               Minimum time spent in a YP Lookup: 0 ms, 1 us
               Maximum time spent in a YP Lookup: 469 ms, 991 us
               Average time spent in YP Lookups: 4 ms, 474 us

                  3 Most Recent Lookups:
              [0] Lookup time: 0 ms, 1 us Number of network re-
transmissions: 0
              [1] Lookup time: 5 ms, 993 us Number of network re-
transmissions: 0
              [2] Lookup time: 0 ms, 1 us Number of network re-
transmissions: 0
NIS netgroup (*.* and *.nisdomain) cache status:
uninitialized
         *.* eCode: 0
         *.nisdomain eCode: 0
   NIS Slave disabled
```

How VLANs work

Traffic from multiple VLANs can traverse a link that interconnects two switches by using VLAN tagging. A VLAN tag is a unique identifier that indicates the VLAN to which a frame belongs. A VLAN tag is included in the header of every frame sent by an end-station on a VLAN.

On receiving a tagged frame, the switch inspects the frame header and, based on the VLAN tag, identifies the VLAN. The switch then forwards the frame to the destination in the identified VLAN. If the destination MAC address is unknown, the switch limits the flooding of the frame to ports that belong to the identified VLAN.



For example, in this figure, if a member of VLAN 10 on Floor 1 sends a frame for a member of VLAN 10 on Floor 2, Switch 1 inspects the frame header for the VLAN tag (to determine the VLAN) and the destination MAC address. The destination MAC address is not known to Switch 1. Therefore, the switch forwards the frame to all other ports that belong to VLAN 10, that is, port 4 of Switch 2 and Switch 3. Similarly, Switch 2 and Switch 3 inspect the frame header. If the destination MAC address on VLAN 10 is known to either switch, that switch forwards the frame to the destination. The end-station on Floor 2 then receives the frame.

How VLAN membership affects communication

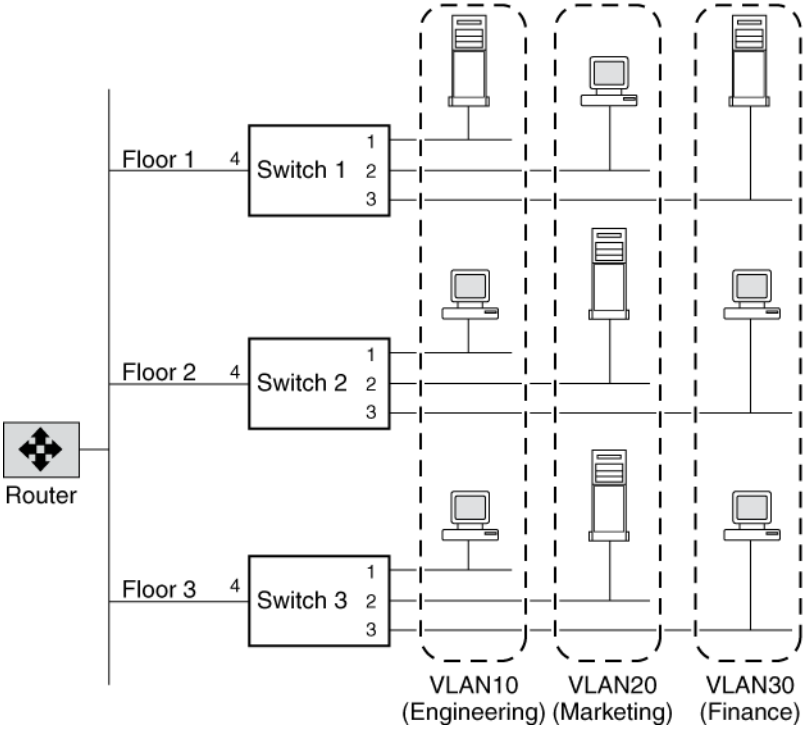
Any broadcast or multicast packets that originate from a member of a VLAN are confined to the members of that VLAN. The communication among VLANs, therefore, must go through a router. A network switch distinguishes among VLANs by associating end-stations to a specific VLAN. This is known as VLAN membership. An end-station must become a member of a VLAN before it can share the broadcast domain with other end-stations on that VLAN.

VLAN membership can be based on one of the following:

- Switch ports
- End-station MAC addresses
- Protocol

In Data ONTAP, VLAN membership is based on switch ports. With port-based VLANs, ports on the same or different switches can be grouped to create a VLAN. As a result, multiple VLANs can exist on a single switch. The switch ports can be configured to belong to one or more VLANs (static registration), or end-stations can register their VLAN membership dynamically, with VLAN-aware switches.

The following figure illustrates how the communication occurs among geographically dispersed VLAN members.



In this figure, VLAN10 (Engineering), VLAN20 (Marketing), and VLAN30 (Finance) span three floors of a building. If a member of VLAN10 on Floor 1 wants to communicate with a member of VLAN10 on Floor 3, the communication occurs without going through the router, and packet flooding is limited to port 1 of Switch 2 and Switch 3 even if the destination MAC address to Switch 2 and Switch 3 is not known.

GARP VLAN Registration Protocol

GARP VLAN Registration Protocol (GVRP) uses Generic Attribute Registration Protocol (GARP) to allow end-stations on a network to *dynamically* register their VLAN membership with GVRP-aware switches. Similarly, these switches dynamically register with other GVRP-aware switches on the network, thus creating a VLAN topology across the network.

GVRP provides dynamic registration of VLAN membership; therefore, members can be added or removed from a VLAN at any time, saving the overhead of maintaining static VLAN configuration on switch ports. Additionally, VLAN membership information stays current, limiting the broadcast domain of a VLAN only to the active members of that VLAN.

For more information about GVRP and GARP, see IEEE 802.1Q and IEEE 802.1p (incorporated in the 802.1D:1998 edition).

GVRP configuration for VLAN interfaces

By default, GVRP is disabled on all VLAN interfaces in Data ONTAP; however, you can enable it.

After you enable GVRP on an interface, the VLAN interface informs the connecting switch about the VLANs it supports. This information (dynamic registration) is updated periodically. This information is also sent every time an interface comes up after being in the down state or whenever there is a change in the VLAN configuration of the interface.

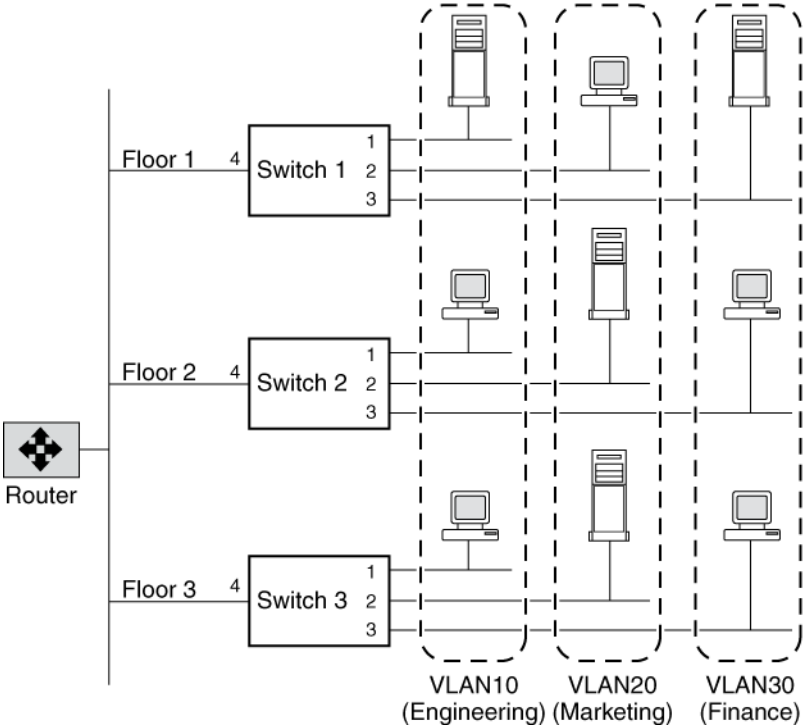
Related tasks

Enabling or disabling GVRP on your VLAN interface on page 92

VLAN tags

A VLAN tag is a unique identifier that indicates the VLAN to which a frame belongs. Generally, a VLAN tag is included in the header of every frame sent by an end-station on a VLAN.

On receiving a tagged frame, the switch inspects the frame header and, based on the VLAN tag, identifies the VLAN. The switch then forwards the frame to the destination in the identified VLAN. If the destination MAC address is unknown, the switch limits the flooding of the frame to ports that belong to the identified VLAN.



For example, in this figure, port 4 on Switch 1, Switch 2, and Switch 3 allows traffic from VLANs 10, 20, and 30. If a member of VLAN 10 on Floor 1 sends a frame for a member of VLAN 10 on Floor 2, Switch 1 inspects the frame header for the VLAN tag (to determine the VLAN) and the destination MAC address. The destination MAC address is not known to Switch 1. Therefore, the switch forwards the frame to all other ports that belong to VLAN 10, that is, port 4 of Switch 2 and Switch 3. Similarly, Switch 2 and Switch 3 inspect the frame header. If the destination MAC address on VLAN 10 is known to either switch, that switch forwards the frame to the destination. The end-station on Floor 2 then receives the frame.

Advantages of VLANs

VLANs provide a number of advantages such as ease of administration, confinement of broadcast domains, reduced network traffic, and enforcement of security policies.

VLANs provide the following advantages:

- Ease of administration
  VLANs enable logical grouping of end-stations that are physically dispersed on a network. When users on a VLAN move to a new physical location but continue to perform the same job function, the end-stations of those users do not need to be reconfigured. Similarly, if users change their job function, they need not physically move: changing the VLAN membership of the end-stations to that of the new team makes the users' end-stations local to the resources of the new team.

- Confinement of broadcast domains
  VLANs reduce the need to have routers deployed on a network to contain broadcast traffic. Flooding of a packet is limited to the switch ports that belong to a VLAN.
- Reduction in network traffic
  Confinement of broadcast domains on a network significantly reduces traffic.
- Enforcement of security policies
  By confining the broadcast domains, end-stations on a VLAN are prevented from listening to or receiving broadcasts not intended for them. Moreover, if a router is not connected between the VLANs, the end-stations of a VLAN cannot communicate with the end-stations of the other VLANs.

Prerequisites for setting up VLANs

You must meet certain prerequisites for switches and end-stations before you can set up VLANs in a network.

The following are the prerequisites for setting up VLANs:

- The switches deployed in the network either must comply with IEEE 802.1Q standards or must have a vendor-specific implementation of VLANs.
- For an end-station to support multiple VLANs, it must be able to dynamically register (using GVRP) or must be statically configured to belong to one or more VLANs.

Guidelines for setting up VLANs in Data ONTAP

VLANs in Data ONTAP are implemented in compliance with the IEEE 802.1Q standard. Before creating and configuring LANs, you should follow certain guidelines about the type, MTU size, speed and media of the underlying interfaces.

The following guidelines apply when setting up VLANs in Data ONTAP:

- You cannot set up VLANs by using the `setup` procedure.
  You must use the command-line interface to create, change, or delete VLANs.
- You must add the commands to create VLANs on the storage system to the `/etc/rc` file to make the VLANs persistent across reboots.
- You can create any number of VLANs on a NIC (supporting IEEE 802.1Q) on the storage system.
  However, Data ONTAP imposes a limit on the number of interfaces (including physical, interface group, VLAN, vh, and loopback interfaces) per storage system.
- The maximum number of VLANs that you can configure is determined by the system memory:
  - For systems with 2-GB memory, you can create a maximum of 128 VLANs.
  - For systems with 6-GB memory, you can create a maximum of 256 VLANs.
  - For systems with greater than 6-GB memory, you can create a maximum of 512 VLANs.

Note: VLAN tags 0 and 1 are reserved and not supported.

- You can create VLANs on physical interfaces and interface groups.
- You can configure IPv4 and IPv6 addresses on a VLAN interface.
- You can use VLANs to support packets of different Maximum Transmission Unit (MTU) sizes on the same network interface.

  If a network interface is a member of multiple VLANs, you can specify different MTU sizes for individual VLANs.
- You can assign an identification number ranging from 1 to 4094 to a VLAN.
- You must ensure that the interface on your storage system is also a member of its partner's VLANs in an HA pair.
- You cannot configure any parameters except `mediatype` for the physical network interface configured to handle VLANs.
- You should set the same flow control settings for all the underlying physical network interfaces that constitute a VLAN.
- You should set the flow control settings of all the network interfaces to `none`.

Related concepts

Maximum number of network interfaces on page 12

Related tasks

Specifying the flow control type for a network interface on page 28

The vlan command syntax

You can use the `vlan` command to create, add interfaces to, delete, modify, and view the statistics of a VLAN.

The following table provides the syntax of the `vlan` command:

Command	Description
`vlan create [-g {on|off}] ifname vlanid_list`	Create a VLAN
`vlan add ifname vlanid_list`	Add an interface to a VLAN
`vlan delete -q ifname [vlanid_list]`	Delete an interface from a VLAN
`vlan modify -g {on|off} ifname`	Enable or disable GVRP on VLAN interfaces
`vlan stat ifname [vlanid_list]`	View the statistics of the network interfaces of a VLAN

For more information about the `vlan` command, see the na_vlan(1) man page.

Note: The VLANs created or changed using the `vlan` command are not persistent across reboots unless the `vlan` commands are added to the `/etc/rc` file.

Creating a VLAN

You can create a VLAN for ease of administration, confinement of broadcast domains, reduced network traffic, and enforcement of security policies. You can use the `vlan create` command to include an interface in one or more VLANs as specified by the VLAN identifier, enable VLAN tagging, and optionally enable GVRP.

About this task

- By default, GVRP is disabled on VLAN interfaces created by using the `vlan create` command; however, you can enable it with the `-g` option of the `vlan create` command.
- VLANs created by using the `vlan create` command are not persistent across reboots unless the `vlan` commands are added to the `/etc/rc` file.
- A VLAN name should not exceed 15 characters. A VLAN is named by combining the base interface name (physical or interface group) and the VLAN identifier. If the resulting VLAN name exceeds 15 characters, the base interface name is truncated and appended to the VLAN identifier with a hyphen (-) in between.
- You should be aware of the limit on the interface name when making an entry in the `/etc/rc` file.

Step

1. Enter the following command:

   `vlan create [-g {on|off}]` *`ifname vlanid`*

   `-g` enables (`on`) or disables (`off`) GVRP on an interface. By default, GVRP is disabled on the interface.

   `ifname` is the name of the network interface.

   `vlanid` is the VLAN identifier to which the *`ifname`* interface belongs. You can include a list of VLAN identifiers.

Example: Creating and naming of VLAN interfaces

Create VLANs with identifiers 10, 20, and 30 on the interface e4 of a storage system by using the following command:

```
vlan create e4 10 20 30
```

As a result, VLAN interfaces e4-10, e4-20, and e4-30 are created. The `ifconfig` command output displays e4 as a VLAN interface as follows:

```
ifconfig -a
e0a: flags=0x80e08866<BROADCAST,RUNNING,MULTICAST,VLAN> mtu 1500
ether 00:0c:29:56:54:7e (auto-1000t-fd-up) flowcontrol full
```

The following example displays the truncation of the base interface name when creating a VLAN. To create a VLAN on the interface group "reallylongname," enter the following command:

`vlan create reallylongname 100`

The resulting VLAN name is "reallylongn-100". The base interface name is truncated and the VLAN name is restricted to 15 characters. When you edit the `/etc/rc` file, ensure that you enter the truncated VLAN name.

After you finish

You must configure the VLAN interface by using the `ifconfig` command.

Related concepts

Prerequisites for setting up VLANs on page 86
Guidelines for setting up VLANs in Data ONTAP on page 86

Configuring a VLAN

After you create a VLAN, you must configure it with an IP address. By using the `ifconfig` command, you can configure all the parameters for a VLAN interface in the same way that you configure the parameters for a physical interface.

About this task

You can configure the following parameters for a VLAN:

- IP address (IPv4 and IPv6)
- Network mask
- Prefix length
- Interface status
- Partner

Step

1. Enter the following command:

   `ifconfig` ***`ifname-vlanid IP_address`*** **`netmask`** ***`mask`***

   `ifname-vlanid` is the VLAN interface name.

   `IP_address` is the IP address for this interface.

mask is the network mask for this interface.

Example

Create VLANs with identifiers 1760 on the interface e5a of a storage system by using the following command:

```
vlan create e5a 1760
```

Configure the VLAN interface e5a-1760 by using the following command:

```
ifconfig e5a-1760 192.0.2.11 netmask 255.255.255.0
```

To configure the VLAN interface e5a-1760 with an IPv6 address, use the following command:

```
ifconfig e5a-1760 2001:0db8:85a3:0:0:8a2e:0370:99 prefixlen 64
```

Related concepts

Configuring network interfaces on page 23

IPv6 link-local addresses for VLANs

When IPv6 is enabled on your storage system, all VLANs share the same link-local address as the underlying network interface (physical or interface group). When VLANs share the same link-local address, there are no address duplication (DAD) issues because link-local addresses cannot be routed and are confined to a VLAN.

Related concepts

IPv6 address scopes on page 45

Related tasks

Enabling or disabling IPv6 on page 47

How to use VLANs for tagged and untagged network traffic

You can configure an IP address for an interface with VLANs. Any untagged traffic goes to the base interface and the tagged traffic goes to the respective VLAN.

You can configure an IP address for the base interface (physical port) of the VLAN. Any tagged frame is received by the matching VLAN interface. Untagged traffic is received by the native VLAN on the base interface.

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.

You cannot bring down the base interface that is configured to receive tagged and untagged traffic. You must bring down all VLANs on the base interface before you bring down the interface. However, you can delete the IP address of the base interface.

When you configure a VLAN using GVRP, the GVRP packets are sent as both untagged and tagged from the native VLAN.

For information about reverting with a configuration for receiving tagged and untagged frames on the same network interface, see the *Data ONTAP Upgrade and Revert/Downgrade Guide for 7-Mode*.

Adding an interface to a VLAN

If a physical interface does not belong to any VLAN, you can use the `vlan create` command to make the interface a member of one or more VLANs. However, if the interface is already a member of a VLAN, you should use the `vlan add` command to add the interface to subsequent VLANs.

About this task

VLANs created using the `vlan add` commands are not persistent across reboots unless the `vlan` commands are added to the `/etc/rc` file.

Step

1. Enter the following command:

   `vlan add` *`interface_name vlanid`*

   `interface_name` is the name of the network interface.

   `vlanid` is the VLAN identifier to which the interface belongs. You can include a list of VLAN identifiers.

   Example
   Add VLANs with identifiers 40 and 50 on the interface e4 of a storage system by using the following command:

   `vlan add e4 40 50`

   VLAN interfaces e4-40 and e4-50 are created.

After you finish

You must configure the VLAN interface by using the `ifconfig` command.

Related tasks

Configuring a VLAN on page 89
Creating a VLAN on page 88

Deleting VLANs

You can delete a specific VLAN or all VLANs that are configured on a network interface. When you delete all VLANs on an interface, the interface is then available to be configured as a regular physical interface.

Step

1. Enter the following command:

   vlan delete [-q] ***interface_name***

If you want to...	Enter the following command:
Delete one or more specific VLANs	**vlan delete [-q]** ***interface_name vlanid*** **Note:** If you want to delete more than one specific VLAN, you can include a list of VLAN identifiers. For example, to delete the VLAN e4-30, enter the following command: **vlan delete e4 30**
Delete all VLANs configured on a network interface	**vlan delete [-q]** ***interface_name*** For example, to delete all VLANs configured on the interface e4, enter the following command: **vlan delete e4**

   interface_name is the name of the network interface.

   vlanid is the VLAN identifier to which the *interface_name* interface belongs. You can include a list of VLAN identifiers.

   `-q` option invokes the quiet mode.

Result

By default, the `vlan delete` command prompts you to confirm the deletion.

Note: If you do not want to receive this prompt, use the `-q` option with the `vlan delete` command. This action invokes the quiet mode, which causes the operation to complete without prompting.

Enabling or disabling GVRP on your VLAN interface

GVRP dynamically registers the VLAN memberships of stations on your network. This reduces the overhead of maintaining static VLAN configuration on switch ports every time a change occurs in

your network. To enable or disable GVRP on all interfaces of a network adapter, you should use the `vlan modify` command.

About this task

- When you enable GVRP on a network interface, it is enabled on all the associated VLANs.
  For example, you can enable GVRP only on the network interface e8 of a storage system. However, you cannot enable or disable GVRP for the VLAN e8-2.
- If you enable GVRP on an interface that is configured to the `down` status, the state of the interface and all associated VLAN interfaces is automatically configured to the `up` status.
  This state change occurs so that the interface can start sending VLAN registration frames to register its VLAN membership with the switch.
- VLANs modified using the `vlan modify` command are not persistent across reboots unless the `vlan` commands are added to the `/etc/rc` file.

Step

1. Enter the following command:

   `vlan modify -g {on|off}` ***`adap_name`***

   `-g on` enables GVRP.

   `-g off` disables GVRP.

   `adap_name` is the name of the network adapter.

Related concepts

GARP VLAN Registration Protocol on page 84
GVRP configuration for VLAN interfaces on page 84

Viewing VLAN statistics

You can use the `vlan stat` command to view the statistics of all VLANs configured on a network interface. You can view the frames received and transmitted on an interface and the number of frames that were rejected because the frames did not belong to any of the VLAN groups.

Step

1. Enter the following command:

   `vlan stat` ***`interface_name`***

   `interface_name` is the name of the network interface.

   #### Example
   The following example displays the statistics of all VLANs on a storage system:

```
vlan stat e4
Vlan Physical Interface: e4 (5 hours, 50 minutes, 38 seconds) --
Vlan IDs: 3,5
GVRP: enabled

RECEIVE STATISTICS
Total frames: 0 | Total bytes: 0 |Multi/broadcast: 0

Untag drops:0 | Vlan tag drops: 0

TRANSMIT STATISTICS
Total frames: 8 | Total bytes: 368

Vlan Interface: e4-3 (0 hours, 20 minutes, 45 seconds) --

ID: 3 MAC Address: 00:90:27:5c:58:14
```

Viewing statistics for a specific VLAN

You can use the `vlan stat` command to view the statistics for a specific VLAN configured on a network interface. You can view the frames received and transmitted on an interface and the number of frames that were rejected because the frames did not belong to any of the VLAN groups.

Step

1. Enter the following command:

   vlan stat *interface_name vlanid*

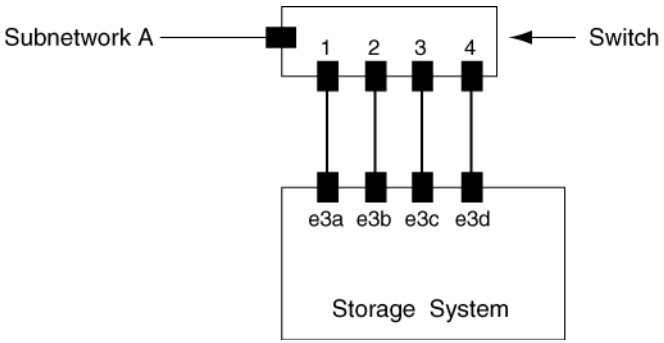
   interface_name is the name of the network interface.

   vlanid is the VLAN identifier to which the *interface_name* interface belongs. You can include a list of VLAN identifiers.

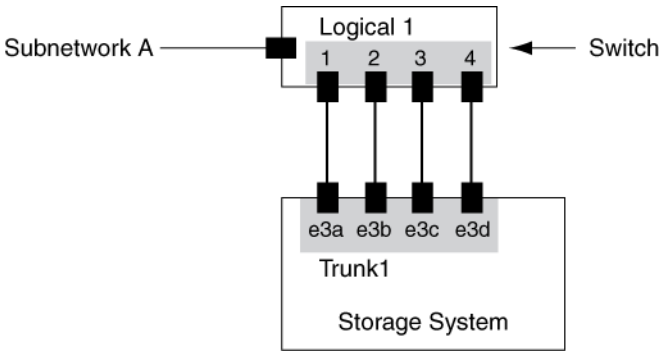
How interface groups work in Data ONTAP

An interface group is a feature in Data ONTAP that implements link aggregation on your storage system. Interface groups provide a mechanism to group together multiple network interfaces (links) into one logical interface (aggregate). After an interface group is created, it is indistinguishable from a physical network interface.

The following figure shows four separate network interfaces, e3a, e3b, e3c, and e3d, before they are grouped into an interface group.



The following figure shows the four network interfaces grouped into a single interface group called Trunk1.



Different vendors refer to interface groups by the following terms:

- Virtual aggregations
- Link aggregations
- Trunks
- EtherChannel

Interface groups provide several advantages over individual network interfaces:

- Higher throughput
  Multiple interfaces work as one interface.
- Fault tolerance
  If one interface in an interface group goes down, your storage system stays connected to the network by using the other interfaces.
- No single point of failure
  If the physical interfaces in an interface group are connected to multiple switches and a switch goes down, your storage system stays connected to the network through the other switches.

Types of interface groups

You can create three different types of interface groups on your storage system: single-mode, static multimode, and dynamic multimode interface groups.

Each interface group provides different levels of fault tolerance. Multimode interface groups provide methods for load balancing network traffic.

Starting with Data ONTAP 7.3.1, IPv6 supports both single-mode and multimode interface groups.

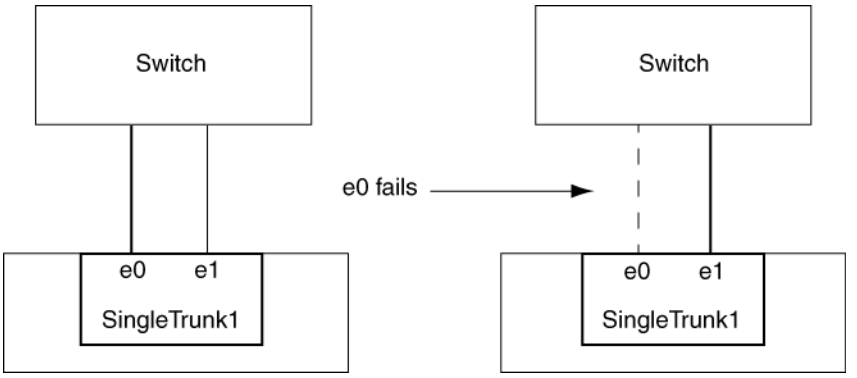
Single-mode interface group

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. All interfaces in a single-mode interface group share a common MAC address.

There can be more than one interface on standby in a single-mode interface group. If an active interface fails, the storage system randomly picks one of the standby interfaces to be the next active link. The active link is monitored and link failover is controlled by the storage system; therefore, single-mode interface group does not require any switch configuration. Single-mode interface groups also do not require a switch that supports link aggregation.

If a single-mode interface group spans over 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 (for example, a LAN or a VLAN). Link-monitoring ARP packets, which have a source address of 0.0.0.0, are sent over the ports of a single-mode interface group to detect whether the ports are in the same broadcast domain.

The following figure is an example of a single-mode interface group. In the figure, e0 and e1 are part of the SingleTrunk1 single-mode interface group. If the active interface, e0, fails, the standby e1 interface takes over and maintains the connection to the switch.



Static multimode interface group

The static multimode interface group implementation in Data ONTAP is in compliance 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 a few characteristics of a static multimode interface group:

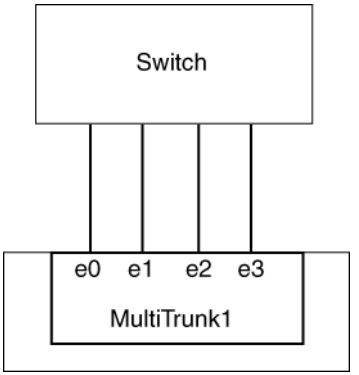
- In a static multimode interface group, all interfaces in the interface group are active and share a single MAC address.

  This logical aggregation of interfaces allows for multiple individual connections to be distributed among the interfaces in the interface group. Each connection or session uses one interface within the interface group and has a reduced likelihood of sharing that single interface with other connections. This effectively allows for greater aggregate throughput, although each individual connection is limited to the maximum throughput available in a single port.
- 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 in a static multimode interface group, the traffic that was traversing that failed link is automatically redistributed to one of the remaining interfaces. If the failed or disconnected port is restored to service, traffic is automatically redistributed among all active interfaces, including the newly restored interface.
- Static multimode interface groups can detect only a loss of link, but cannot detect a situation where data cannot flow to or from the directly attached network device (for example, a switch).
- 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 e0, e1, e2, and e3 are part of the MultiTrunk1 multimode interface group. All four interfaces in the MultiTrunk1 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 Data ONTAP conform to the IEEE 802.3 standards. If a particular multiple switch link aggregation technology is stated to interoperate or conform to the IEEE 802.3 standards, it should operate with Data ONTAP.

The IEEE 802.3 standard states that the transmitting device in an aggregated link determines the physical interface for transmission. Therefore, Data 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, it must be modified on the directly connected network device.

Dynamic multimode interface group

Dynamic multimode interface groups 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 storage controller to communicate with the direct-attached switch port.

Dynamic multimode interface group implementation in Data ONTAP is in compliance with IEEE 802.3 AD (802.1 AX). Data 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.

Data ONTAP implements LACP in nonconfigurable active mode that works well with switches that are configured in either active or passive mode. Data 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 Data 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 switch vendor's documentation.

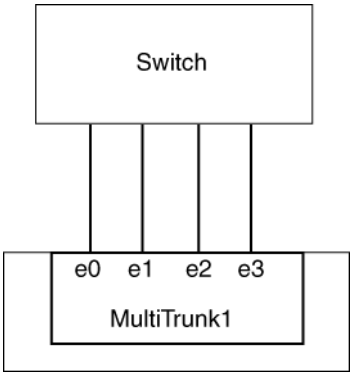
The following rules apply when using dynamic multimode interface groups:

- You must configure the dynamic multimode interface groups as first-level 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 network interfaces and the switch ports that are members of a dynamic multimode interface groups must be set to use the same speed, duplex, and flow control settings.

  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 re-routed to any remaining active interfaces.

  Note: Some switches might not support link aggregation of ports configured for jumbo frames.

The following figure is an example of a dynamic multimode interface group. Interfaces e0, e1, e2, and e3 are part of the MultiTrunk1 multimode interface group. All four interfaces in the MultiTrunk1 dynamic multimode interface group are active.



Related references

What the interface group status information table contains on page 111

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, round-robin, 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 storage 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.

Round-robin load balancing

You can use round-robin for load balancing multimode interface groups. You should use the round-robin 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.

Guidelines for configuring interface groups

Before creating and configuring interface groups, you should follow certain guidelines about the type, MTU size, speed, and media of the underlying interfaces.

The following guidelines apply when you create and configure interface groups on your storage system:

- The network interfaces that are part of an interface group should be on the same network adapter.
- You can configure a maximum of eight network interfaces in a single interface group.
- You cannot include a VLAN interface in an interface group.
- The interfaces that form an interface group must have the same Maximum Transmission Unit (MTU) size.

  If you attempt to create or add to an interface group and the member interfaces have different MTU sizes, Data ONTAP automatically modifies the MTU size to be the same. To ensure that the desired MTU size is configured, you can use the `ifconfig` command to configure the MTU size of the interface group after it is created. You need to configure the MTU size only if you are enabling jumbo frames on the interfaces.
- You can include any interface, except the e0M management interface that is present on some storage systems.
- You should not mix interfaces of different speeds or media in the same multimode interface group.
- You should set the same flow control settings for all the underlying physical network interfaces that constitute an interface group.
- You should set the flow control settings of all the network interfaces to `none`.

Some switches might not support multimode link aggregation of ports configured for jumbo frames. For more information, see your switch vendor's documentation.

Related tasks

Specifying the flow control type for a network interface on page 28

The ifgrp command

You can manage interface groups on your storage system by using the `ifgrp` command. This command enables you to create interface groups, add interfaces to interface groups, delete interfaces from interface groups, view status and statistics of interface groups, and destroy interface groups.

The following table provides the `ifgrp` command syntax:

Command	Description
ifgrp create [single|multi|lacp] *ifgrp_name* -b [rr|mac|ip] [*interface_list*]	Create a single-mode or multimode interface group
`ifgrp {favor|nofavor}` *interface_name*	Designate a favored or nonfavored interface in a single-mode interface group
`ifgrp add` *ifgrp_name interface_list*	Add network interfaces to an interface group
`ifgrp delete` *ifgrp_name interface_name*	Delete a network interface from an interface group
`ifgrp destroy` *ifgrp_name*	Destroy an interface group
`ifgrp status` [*ifgrp_name*]	View the status of an interface group
`ifgrp stat` *ifgrp_name* [*interval*]	View the statistics of data packets on the network interfaces of an interface group

The following `ifgrp` commands are not persistent if used from the command-line interface; however, you can put any of these commands in the `/etc/rc` file to make it persistent across reboots:

- `ifgrp create`
- `ifgrp add`
- `ifgrp delete`
- `ifgrp destroy`
- `ifgrp favor`
- `ifgrp nofavor`

Creating a single-mode interface group

You can create a single-mode interface group in which only one interface is active at a time and the others are ready to take over if the active interface fails. A single-mode interface group increases the redundancy for providing more availability.

Before you begin

- Decide on a case-sensitive name for the interface group that meets the following criteria:
  - It must begin with a letter.
  - It must not contain any spaces.
  - It must not contain more than 15 characters.
  - It must not already be in use for an interface group.

- Decide on a list of the interfaces you want to combine into the interface group.
- To make a specific interface active, you must specify that interface as preferred by using the `ifgrp favor` command; otherwise, an interface is randomly selected to be the active interface.

Steps

1. Configure all interfaces that are to be included in the interface group to the `down` status by entering the following command:

   ifconfig *interface_list* down

   interface_list is a list of the interfaces you want as part of the interface group.

   Example

   ifconfig e0a e0b down

2. To create an interface group, enter the following command:

   ifgrp create single *ifgrp_name* [*interface_list*]

   ifgrp_name is the name of the interface group.

   interface_list is a list of the interfaces you want as part of the interface group.

   Note: The operation performed using the `ifgrp create` command is not persistent across reboots unless you add the command to the `/etc/rc` file.

3. To configure the interface group, enter the following command:

   ifconfig *ifgrp_name IP_address*

   ifgrp_name is the name of the interface group.

   IP_address is the IP address for this interface.

   Note: If you have enabled IPv6 on your storage system, you can create an interface group and then configure the interface group to the `up` status. After this, the interface group has two IPv6 addresses automatically configured on it. Therefore, you need not manually configure the IP address for an interface group.

Example: Creating a single-mode interface group with an IPv4 address

The following command creates a single-mode interface group SingleTrunk1:

ifgrp create single SingleTrunk1 e0 e1

The following command configures an IP address of 192.0.2.4 and a netmask of 255.255.255.0 on the single-mode interface group SingleTrunk1:

ifconfig SingleTrunk1 192.0.2.4 netmask 255.255.255.0

Example: Creating a single-mode interface group when IPv6 is enabled

The following command creates a single-mode interface group:

ifgrp create single SingleTrunk1 e0 e1

You can configure the interface group by using one of the following methods:

- You can automatically configure an IPv6 address for the interface group by configuring the interface to the up status with the following command:

  ifconfig SingleTrunk1 up

  The following example output shows two automatically configured addresses for the interface group:

  ```
  system1> ifconfig SingleTrunk1
  SingleTrunk1: flags=0x20608862<UP,BROADCAST,RUNNING,MULTICAST,TCPCKSUM> mtu 1500
  inet6 2001:0db8:a0:98ff:fe08:618a prefixlen 64 scopeid 0x9 autoconf
  inet6 2001:0db8:a0:98ff:fe08:618a prefixlen 64 autoconf
  ether 02:a0:98:08:61:8a (Enabled virtual interface)
  ```

- You can manually configure an IPv6 address of 2001:0db8:85a3:0:0:8a2e:0370:99 for the interface group by using the following command:

  ifconfig SingleTrunk1 2001:0db8:85a3:0:0:8a2e:0370:99

Related concepts

Single-mode interface group on page 96

Related tasks

Changing the status of an interface on page 34

Selecting an active interface in a single-mode interface group

When you create a single-mode interface group, an interface is randomly selected to be the active interface. You can specify another interface as active—for example, when you add a higher speed or higher bandwidth interface—by using the `ifgrp favor` command to override the random selection.

Step

1. Enter the following command:

   ifgrp favor *interface_name*

   interface_name is the name of the interface that you want to specify as active.

   Example

   To specify the interface e1 as preferred, enter the following command:

   ifgrp favor e1

   Note: The operation performed using the `ifgrp favor` command is not persistent across reboots unless the command is added to the `/etc/rc` file.

Related concepts

Single-mode interface group on page 96

Related tasks

Designating a nonfavored interface in a single-mode interface group on page 105

Designating a nonfavored interface in a single-mode interface group

When you create a single-mode interface group, an interface is randomly selected to be the active interface. You can designate an interface as nonfavored so that it is not considered during the random selection of an active interface in a single-mode interface group.

About this task

The interface marked as nonfavored can become the active interface when all other interfaces in a single-mode interface group fail. Even after other interfaces come to the `up` state, a nonfavored interface continues to remain the active interface until it fails or until you, the system administrator, change the active interface by using the `ifgrp favor` command.

Step

1. Enter the following command:

   `ifgrp nofavor` ***`interface_name`***

   `interface_name` is the name of the interface you do not want to be considered during the random selection of an active interface.

   Note: The operation performed using the `ifgrp nofavor` command is not persistent across reboots unless the command is added to the `/etc/rc` file.

   Example

   Specify the interface e2 to be nonfavored with the following command:

   ```
   ifgrp nofavor e2
   ```

Related concepts

Single-mode interface group on page 96

Related tasks

Selecting an active interface in a single-mode interface group on page 104

Failure scenarios for a single-mode interface group

A single-mode interface group fails when the link status of the interface group is `down`. Failure can also occur if link-monitoring Address Resolution Protocol (ARP) packets do not reach any of the interfaces that form the interface group.

When the link status of a single-mode interface group is configured to the `down` status, it signals that the interfaces that are part of the interface group have lost connection with the switch.

Link-monitoring ARP packets are sent over the ports of a single-mode interface group to detect whether the ports are in the same broadcast domain. If these ARP packets do not reach any of the interfaces in the interface group, the interface group is configured to the `down` status. To avoid this problem, you must ensure that all the interfaces of a single-mode interface group are in the same broadcast domain (for example, a LAN or a VLAN).

Related concepts

Single-mode interface group on page 96

Related tasks

Viewing interface group status on page 111

Creating a static multimode interface group

You can use the `ifgrp create` command to create a static multimode interface group. If you do not specify the type of interface group in the `ifgrp create` command, a static multimode interface group is created by default.

Before you begin

- A switch that supports link aggregation over multiple port connections in your network, configured according to your switch vendor's instructions, must be installed.
- A case-sensitive name for the interface group that meets the following criteria must be decided:
  - It must begin with a letter.
  - It must not contain a space.
  - It must not contain more than 15 characters.
  - It must not already be in use for an interface group.
- The interfaces that you want to select as part of the interface group should be decided.
- All the interfaces that are to be included in the interface group should be configured to the `down` status, by using the `ifconfig` command.

About this task

You can improve throughput by creating a static multimode interface group. With a multimode interface group, all interfaces in the interface group are active and share a single MAC address. This logical aggregation of interfaces provides higher throughput than a single-mode interface group.

Steps

1. To create the interface group, enter the following command:

   ifgrp create multi *ifgrp_name* -b {rr|mac|ip|port} [*interface_list*]

   `-b` describes the load-balancing method.

   `rr` specifies round-robin load balancing.

   `mac` specifies MAC address load balancing.

   Note: Do not select the MAC address load-balancing method when creating interface groups on a storage 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` indicates IP address load balancing (default).

   `port` indicates port-based load balancing.

   `ifgrp_name` is the name of a previously created interface group.

   `interface_list` is a list of the interfaces you want to add to the interface group.

   Example
   To create a static multimode interface group, comprising interfaces e0, e1, e2, and e3 and using MAC address load balancing, enter the following command:

   ifgrp create multi MultiTrunk1 -b mac e0 e1 e2 e3

2. To configure the interface group, enter the following command:

   ifconfig *ifgrp_name IP_address* netmask *mask*

Related concepts

Static multimode interface group on page 97
Load balancing in multimode interface groups on page 100

Related tasks

Changing the status of an interface on page 34

Creating a dynamic multimode interface group

With a dynamic multimode interface group, all interfaces in the interface group are active and share a single MAC address. This logical aggregation of interfaces provides higher throughput than a single-

mode interface group. Dynamic multimode interface groups can detect both loss of link and loss of data flow.

Before you begin

You must meet the following prerequisites to create a multimode interface group:

- Identify or install a switch that supports LACP over multiple port connections in your network, configured according to your switch vendor's instructions.
- Decide on a case-sensitive name for the interface group that meets the following criteria:
  - It must begin with a letter.
  - It must not contain a space.
  - It must not contain more than 15 characters.
  - It must not already be in use for an interface group.
- Decide on the interfaces you want to select as part of the interface group.
- Configure all interfaces that are to be included in the interface group to the `down` status, by using the `ifconfig` command.

About this task

Data ONTAP logs information about the LACP negotiation for dynamic multimode interface groups in the `/vol0/etc/log/lacp_log` file.

Steps

1. To create a dynamic multimode interface group, enter the following command:

   `ifgrp create lacp` *`ifgrp_name`* `-b {rr|mac|ip|port} [`*`interface_list`*`]`

   `-b` specifies the load-balancing method.

   `rr` specifies round-robin load balancing.

   `mac` specifies MAC address load balancing.

   Note: Do not select the MAC address load-balancing method when creating interface groups on a storage 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` specifies IP address based load balancing (default).

   `port` indicates port-based load balancing.

   `ifgrp_name` is the name of a previously created interface group.

   `interface_list` is a list of the interfaces that form the interface group.

Example

To create a dynamic multimode interface group, comprising interfaces e0, e1, e2, and e3 and using IP address based load balancing, enter the following command:

```
ifgrp create lacp MultiTrunk1 -b ip e0 e1 e2 e3
```

2. To configure the dynamic multimode interface group, enter the following command:

   ```
   ifconfig ifgrp_name IP_address netmask mask
   ```

Related concepts

Dynamic multimode interface group on page 98
Load balancing in multimode interface groups on page 100

Related tasks

Changing the status of an interface on page 34

Adding interfaces to an interface group

You can add one or more interfaces to an interface group to expand your network. You can add physical interfaces to an interface group any time after you create it by using the `ifgrp add` command.

Before you begin

- You must configure additional ports on the switch where the new interfaces will be connected. For information about configuring the switch, see your switch vendor's documentation.
- The interface to be added to the interface group must be configured to the `down` status by using the `ifconfig` command.

Step

1. Enter the following command:

   ```
   ifgrp add ifgrp_name interface_list
   ```

   `ifgrp_name` is the name of a previously configured interface group.

   `interface_list` is a list of the interfaces you want to add to the interface group.

   Note: The operation performed using the `ifgrp add` command is not persistent across reboots unless the command is added to the `/etc/rc` file.

Example

To add the interface e4 to the multimode interface group MultiTrunk1, enter with the following command:

```
ifgrp add MultiTrunk1 e4
```

Related tasks

Changing the status of an interface on page 34

Deleting interfaces from an interface group

You might have to delete a physical interface from an interface group—for example, when the interface needs maintenance or when you want to use the interface for a different purpose. You can use the `ifgrp delete` command to delete one or more interfaces from an interface group.

Before you begin

You must configure the interface group to the `down` state before you can delete a network interface from the interface group. You can configure the interface group to the `down` status by using the following command:

`ifconfig` *`ifgrp_name`* `down`

`ifgrp_name` is the name of the interface group that you want to configure to the `down` status.

About this task

The operation performed using the `ifgrp delete` command is not persistent across reboots unless the command is added to the `/etc/rc` file.

Step

1. Enter the following command:

   `ifgrp delete` *`ifgrp_name interface`*

   `ifgrp_name` is the name of the interface group.

   `interface` is the interface of the interface group you want to delete.

   Example
   To delete the interface e4 from a multimode interface group MultiTrunk1, enter the following commands:

   `ifconfig MultiTrunk1 down`

   `ifgrp delete MultiTrunk1 e4`

Related tasks

Changing the status of an interface on page 34

Viewing interface group status

You can view the current status of a specified interface group or all single-mode and multimode interface groups on your storage system.

Step

1. Enter the following command:

   ifgrp status [*ifgrp_name*]

   ifgrp_name is the name of the interface group whose status you want to display.

   If you do not specify the interface group name, the status of all interface groups is displayed.

Example

The following example displays the status of the interface group ifgrp1:

```
ifgrp status ifgrp1
default: transmit 'IP Load balancing', Ifgrp Type 'multi_mode', fail 'log'
ifgrp1: 1 link, transmit 'none', Ifgrp Type 'single_mode' fail 'default'
         Ifgrp Status   Up      Addr_set
        up:
        e0b: state up, since 23Jun2009 08:18:12 (00:01:16)
                mediatype: auto-1000t-fd-up
                flags: enabled
                input packets 54, input bytes 4858
                output packets 35, output bytes 2830
                output probe packets 0, input probe packets 0
                strike count: 0 of 10
                up indications 1, broken indications 0
                drops (if) 0, drops (link) 0
                indication: up at 23Jun2009 08:18:12
                        consecutive 75, transitions 1
        broken:
        e0c: state broken, since 23Jun2009 08:18:22 (00:01:06)
                mediatype: auto-unknown-down
                flags:
                input packets 0, input bytes 0
                output packets 0, output bytes 0
                output probe packets 0, input probe packets 0
                strike count: 0 of 10
                up indications 0, broken indications 0
                drops (if) 0, drops (link) 0
                indication: broken at 23Jun2009 08:18:22
                        consecutive 0, transitions 1
```

What the interface group status information table contains

You can view the status information of an interface group by using the `ifgrp status` command.

The following table describes the information that is shown in each field and subfield of the `ifgrp status` command output.

Field	Subfield	Description
`default`		Indicates the default values for fields such as transmit, Ifgrp Type, and fail. These values apply if you do not specify any values for these fields when creating an interface group.
	`transmit`	Indicates the default load-balancing method.
	`Ifgrp Type`	Indicates the default interface group type.
	`fail`	Indicates the default location where the errors are logged.
`ifgrp_name`		Indicates that the data that follows this field pertains to the interface group, *`ifgrp_name`*.
	`transmit`	Indicates the load-balancing method used.
	`Ifgrp Type`	Indicates the type of interface group. Possible values are single-mode, multi_mode, or lacp.
	`lag_inactive`	Indicates that the network interface belonging to the lacp group is down
	`fail`	Indicates the location where errors are logged for the interface group.
	`Ifgrp Status`	Indicates the current status of the interface group, *`ifgrp_name`*.
	`Addr_set`	Indicates that a MAC address has been configured for the interface group, *`ifgrp_name`*, and all its interfaces.
	`state`	Indicates the current link-state of the interface. Possible values are up or down.
	`since`	Indicates the date, time, and number of hours since the interface has been up.
	`mediatype`	Indicates the media type that defines the speed and duplex for that interface.
	`flags`	Indicates whether the interface is enabled or disabled to send and receive data.

Field	Subfield	Description
	`strike count`	Indicates the number of attempts for link-monitoring. When an underlying link of an interface group does not receive any packets (including ARP packets that are used for link-monitoring), the strike count gets incremented once in 5 seconds. If this strike count reaches 10, the underlying link is brought "down."
	`consecutive`	Indicates the number of consecutively received "up" or "broken" indications from the switch and link interaction.
	`transitions`	Indicates the number of indications received that caused a state transition from "up" to "broken" or "down" to "up".

Viewing interface group statistics

You can view the statistics for a specific interface group or for all interface groups. The statistics include the number of packets received and sent by each interface group.

Step

1. Enter the following command:

   `ifgrp stat [`*`ifgrp_name`*`] [`*`interval`*`]`

   `ifgrp_name` is the name of the interface group. If you do not specify an interface group, the status of all interface groups is displayed.

   `interval` is the interval, in seconds. The default is one second.

Example

The following example displays the output of the `ifgrp stat` command for a multimode interface group created with the round-robin load-balancing method:

```
ifgrp stat ifgrp0
ifgrp (trunk) ifgrp0
         e3a                    e3b
  Pkts In    Pkts Out   Pkts In    Pkts Out
  8637076    47801540   158        159
  1617       9588       0          0
  1009       5928       0          0
  1269       7506       0          0
  1293       7632       0          0
  920        5388       0          0
  1098       6462       0          0
```

```
  2212       13176      0          0
  1315       7776       0          0
```

The first row of the output shows the total number of packets received and sent until the time the `ifgrp stat` command was run. The following rows show the total number of packets received and sent per second thereafter.

For interface groups created with the round-robin load-balancing option, the outgoing packets are balanced among the network interfaces of the interface group.

```
ifgrp stat ifgrp1
Interface group (trunk) ifgrp1
        e0c                 e0b
  Pkts In   Pkts Out  Pkts In   Pkts Out
  82        208k      796k      208k
  1         27342     104774    27326
  2         26522     102088    26560
  8         20332     77275     20335
  5         27198     103529    27186
```

Destroying an interface group

You destroy an interface group when you no longer need it or when you want to use the interfaces that form the interface group for other purposes. After you destroy the interface group, the interfaces in the interface group act individually rather than as an aggregate.

Steps

1. Configure the interface group to the `down` status by entering the following command:

   ifconfig *ifgrp_name* down

   ifgrp_name is the name of the interface group you want to configure to the `down` status.

2. Enter the following command:

   ifgrp destroy *ifgrp_name*

   ifgrp_name is the name of the interface group you want to destroy.

Second-level interface groups

If you have more than one multimode interface group, you can use the `ifgrp create` command to group them by creating a second layer of interface group called the *second-level interface group*. Second-level interface groups enable you to provide a standby multimode interface group in case the primary multimode interface group fails.

You can use second-level interface groups on a single storage system or in an HA pair.

Note: You cannot use LACP interface groups as second-level interface groups.

Guidelines for creating a second-level interface group

You can create a single-mode second-level interface group over two multimode interface groups. The ports of the underlying multimode interface groups should be connected to the same switch. If you create a second-level interface group over two multimode interface groups that are connected to two different switches, you must connect the two switches with an inter-switch link (ISL).

For a single-mode interface group, the switch ports must be in the same broadcast domain (for example, a LAN or a VLAN). Link-monitoring ARP packets, which have a source address of 0.0.0.0, are sent over the ports of a single-mode interface group to detect whether the ports are in the same broadcast domain. If the ports are not in the same broadcast domain and if no data traffic is detected over the ports, the interface group is configured to the `down` status.

When the ports of a single-mode interface group are connected to different broadcast domains, it is called a *split-network condition*, which is difficult to diagnose for problems. To avoid such a situation, if a second-level interface group over two multimode interface groups are connected to two different switches, you must connect the switches by using an ISL.

Creating a second-level interface group

You can create a second-level interface group by using two multimode interface groups. Second-level interface groups enable you to provide a standby multimode interface group in case the primary multimode interface group fails.

Before you begin

You must meet the following prerequisites to create a second-level interface group:

- Identify or install a switch that supports link aggregation over multiple port connections in your network, configured according to your switch vendor's instructions.
- Decide on a name for the second-level interface group:
  - It must begin with a letter.
  - It must not contain a space.
  - It must not contain more than 15 characters.
  - It must not already be in use for an interface group.
- Decide on a list of the interfaces you want to select as part of the interface group.
- Configure all interfaces that are to be included in the interface group to the `down` status, by using the `ifconfig` command.

Steps

1. Enter the following command to create the first of two multimode interfaces:

   `ifgrp create multi -b {rr|mac|ip|port}` ***`ifgrp_name1 if1 if2`***

   The *`ifgrp_name1`* interface group is composed of two physical interfaces, *`if1`* and *`if2`*.

`-b`—specifies the type of load-balancing method.

`rr`—specifies the round-robin load-balancing option.

`mac`—specifies the MAC address load-balancing option.

`ip`—indicates the IP address load-balancing option (default option).

`port`—indicates the port-based load-balancing option.

2. Enter the following command to create the second of two multimode interfaces:

   ifgrp create multi -b {rr|mac|ip|port} *ifgrp_name2 if3 if4*

   The *`ifgrp_name2`* interface group is composed of two physical interfaces, *`if3`* and *`if4`*.

3. Enter the following command to create a single-mode interface from the multimode interfaces:

   ifgrp create single *ifgrp_name ifgrp_name1 ifgrp_name2*

   `ifgrp_name` is the second-level interface group created with two multimode interface groups, *`ifgrp_name1`* and *`ifgrp_name2`*.

Example

Use the following commands to create two interface groups and a second-level interface group. In this example, IP address load balancing is used for the multimode interface groups.

```
ifgrp create multi Firstlev1 e0 e1
ifgrp create multi Firstlev2 e2 e3
ifgrp create single Secondlev Firstlev1 Firstlev2
```

Related tasks

Changing the status of an interface on page 34

Enabling failover in a second-level interface group

An interface group can experience link failures and become degraded. If a different interface group has a higher aggregate bandwidth, failover to this group occurs. If, however, the degraded interface is favored through use of the `favor` option, failover does not occur. Enabling the `ifgrp.failover.link_degraded` option allows failover to occur.

About this task

The `ifgrp.failover.link_degraded` option applies to configurations of a second-level, single-mode interface group containing two or more multimode interface groups where one of the groups has been set to favored by using the `favor` option.

- If `ifgrp.failover.link_degraded` is set to `on` and one or more links in the active, favored multimode interface group fails, a failover to a multimode interface group that has a higher aggregate bandwidth occurs.
- If this option is set to `off`, the default setting, no failover occurs and the favored, degraded interface remains active.

Step

1. If an active, favored interface group becomes degraded, enable failover to an interface group with a higher aggregate bandwidth by entering the following command:

   `options ifgrp.failover.link_degraded on`

Example

In the following scenarios, a second-level, single-mode interface group over two multimode interface groups has the following configuration settings:

- ifgrp1: Multimode interface group over two 1-GbE interfaces
- ifgrp2: Multimode interface group over two 1-GbE interfaces

The possible scenarios and failover behaviors are as follows:

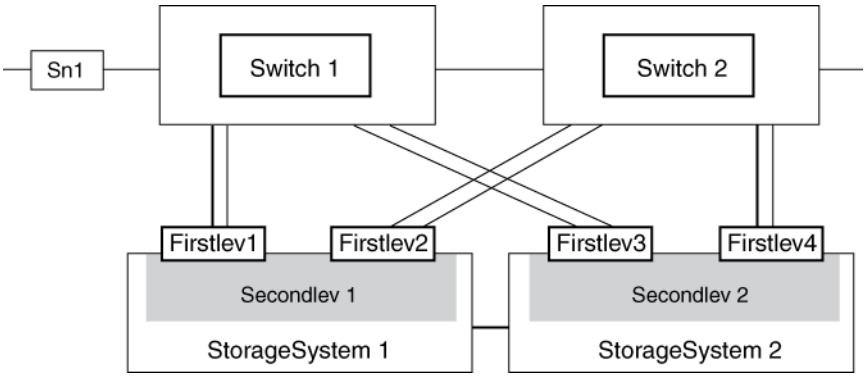
- ifgrp1 is active and neither interface group is favored:
  If one of the links in ifgrp1 fails and ifgrp2 has a higher aggregate bandwidth, ifgrp2 becomes the active interface group. This happens regardless of whether `ifgrp.failover.link_degraded` is set to `on` or `off`.
- ifgrp1 is both active and favored and `ifgrp.failover.link_degraded` is set to `off`:
  If one of the links in ifgrp1 fails, ifgrp1 continues to be the active interface group even if ifgrp2 has a higher aggregate bandwidth.
- ifgrp1 is both active and favored and `ifgrp.failover.link_degraded` is set to `on`:
  If one of the links in ifgrp1 fails and ifgrp2 has a higher aggregate bandwidth, ifgrp2 becomes the active interface group. If ifgrp1 has the higher aggregate bandwidth, it remains active.

Second-level interface groups in an HA pair

In an HA pair, you can access data from both storage systems even if one of the storage system in the configuration fails.

With a second-level interface group connected in a single-mode configuration, you can maintain connectivity to your storage system even if one of the switches fails. Therefore, by using the two configurations together, you can achieve a fully redundant storage system connectivity architecture.

The following figure shows second-level interface groups in an HA pair.



When both storage systems are in operation, the following connections exist:

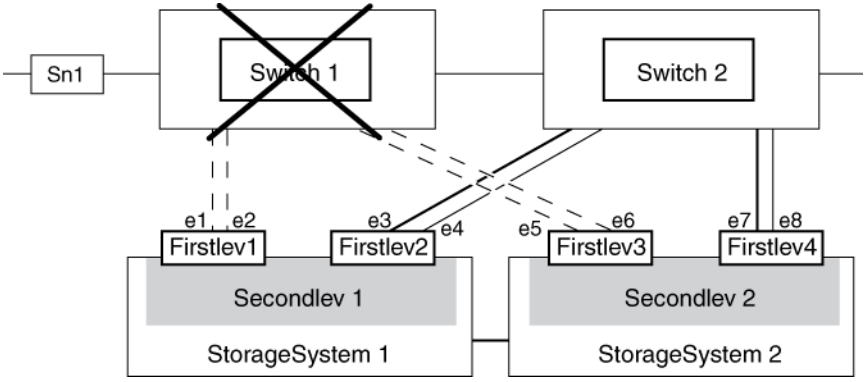
- Firstlev1 in Secondlev 1 connects StorageSystem 1 to the network through Switch 1.
- Firstlev2 in Secondlev 1 connects StorageSystem 1 to Switch 2.
- Firstlev4 in Secondlev 2 connects StorageSystem 2 to the network through Switch 2.
- Firstlev3 in Secondlev 2 connects StorageSystem 2 to Switch 1.

Firstlev2 and Firstlev3 are in standby mode.

If one of the switches fails, the following happens:

- If Switch 1 fails, Firstlev2 and Firstlev4 maintain the connection for their storage systems through Switch 2.
- If Switch 2 fails, Firstlev1 and Firstlev3 maintain the connection for their storage systems through Switch 1.

In the following figure, Switch 1 fails in an HA pair. Firstlev2 takes over the MAC address of Firstlev1 and maintains the connectivity through Switch 2.



Creating a second-level interface group in an HA pair

You can create two second-level interface groups in an HA pair so that you can access data from both storage systems even if one of the storage system in the configuration fails.

Before you begin

You must ensure that all interfaces to be included in the interface group are configured to the `down` status. You can use the `ifconfig` command to configure an interface to the `down` status.

About this task

The operation performed using the `ifgrp create` command is not persistent across reboots unless the command is added to the `/etc/rc` file.

Assume *`StorageSystem1`* and *`StorageSystem2`* are the storage systems that are configured in an HA pair.

Steps

1. Enter the following commands on *`StorageSystem1`* to create two multimode interface groups:

   `ifgrp create multi -b {rr|mac|ip|port}` ***`ifgrp_name1 if1 if2`***

   `ifgrp create multi -b {rr|mac|ip|port}` ***`ifgrp_name2 if3 if4`***

   `-b` specifies the type of load-balancing method.

   `rr` specifies the round-robin load-balancing option.

   `mac` specifies the MAC address load-balancing option.

   `ip` specifies the IP address load-balancing option (default option).

   `port` specifies the port-based load-balancing option.

   `if1`, *`if2`*, *`if3`*, *`if4`* are the network interfaces.

   `ifgrp_name1` and *`ifgrp_name2`* are the names of the multimode interface groups.

2. Enter the following command on *`StorageSystem1`* to create a second-level interface from the multimode interface groups:

   `ifgrp create single` ***`secondlev1 ifgrp_name1 ifgrp_name2`***

   `secondlev1` is the name of the second-level interface group.

3. Enter the following commands on *`StorageSystem2`* to create two multimode interface groups:

   `ifgrp create multi -b {rr|mac|ip|port}` ***`ifgrp_name3 if5 if6`***

   `ifgrp create multi -b {rr|mac|ip|port}` ***`ifgrp_name4 if7 if8`***

4. Enter the following command on *StorageSystem2* to create a second-level interface from the multimode interface groups:

   ifgrp create single *secondlev2 ifgrp_name3 ifgrp_name4*

5. Enter the following command on *StorageSystem1* to configure the second-level interface groups for takeover:

   ifconfig *secondlev1* partner *secondlev2*

6. Enter the following command on *StorageSystem2* to configure the second-level interface groups for takeover:

   ifconfig *secondlev2* partner *secondlev1*

   In steps 5 and 6, secondlev1 and secondlev2 (arguments to the partner option) must be interface names and not interface IP addresses. If secondlev1 is an interface group, secondlev2 can be an interface group or a physical network interface.

Example

Use the following commands to create a second-level interface group in an HA pair. In this example, IP-based load balancing is used for the multimode interface groups.

On *StorageSystem1*:

```
ifgrp create multi Firstlev1 e1 e2
ifgrp create multi Firstlev2 e3 e4
ifgrp create single Secondlev1 Firstlev1 Firstlev2
```

On *StorageSystem2* :

```
ifgrp create multi Firstlev3 e5 e6
ifgrp create multi Firstlev4 e7 e8
ifgrp create single Secondlev2 Firstlev3 Firstlev4
```

On *StorageSystem1*:

```
ifconfig Secondlev1 partner Secondlev2
```

On *StorageSystem2* :

```
ifconfig Secondlev2 partner Secondlev1
```

Related tasks

Changing the status of an interface on page 34
Configuring a partner interface in an HA pair on page 30

How CDP works with Data ONTAP

In a data center, you can use Cisco Discovery Protocol (CDP) to view network connectivity between a pair of physical or virtual systems and their network interfaces. CDP is also useful for verifying network connectivity before performing online migration of vFiler units.

CDP enables you to automatically discover and view information about directly connected CDP-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 CDP-enabled devices.

Neighboring devices of the storage system that are discovered by using CDP are called *CDP neighbors*. For two devices to become CDP neighbors, each must have the CDP protocol enabled and correctly configured. The functionality of CDP is limited to directly connected networks. CDP neighbors include CDP-enabled devices such as switches, routers, bridges, and so on.

Considerations for using CDP

By default, Cisco devices or CDP-compliant devices send CDPv2 advertisements. CDP-compliant devices send CDPv1 advertisements only when they receive CDPv1 advertisements. Data ONTAP supports only CDPv1. Therefore, when the storage system sends CDPv1 advertisements, the CDP-compliant neighboring devices send back CDPv1 advertisements.

You should consider the following information before enabling CDP on your storage system:

- CDP advertisements are sent only by the ports that are in the `up` state.

  In Data ONTAP 8.0.1 and later, the network interface need not be configured with an IP address to be able to send the CDP advertisement.
- 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 at the storage system side, the storage system sends the updated information in the next CDP advertisement.

  Note: Sometimes when IP addresses are changed at the storage system side, 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 storage system 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 the IP addresses configured on the VLANs on that port are advertised.
- For physical ports that are part of an interface group, all the IP addresses configured on that interface group are advertised on each physical port.

- For an interface group that hosts VLANs, all the IP addresses 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 1500 bytes, only the number of IP addresses that can fit into a 1500 MTU-sized packet are advertised.

Enabling or disabling CDP

To discover and send advertisements to CDP-compliant neighboring devices, CDP must be enabled on the storage system. You can use the `cdpd.enable` option to enable or disable CDP.

About this task

When the `cdpd.enable` option is set to `on`, CDPv1 is enabled on all physical ports of the storage system.

Step

1. To enable or disable CDP, enter the following command:

   `options cdpd.enable {on|off}`

   `on`—Enables CDP

   `off`—Disables CDP

Configuring hold time for CDP messages

Hold time is the period of time for which all CDP advertisements are stored in a cache in the neighboring CDP-compliant devices. Hold time is advertised in each CDPv1 packet and is updated whenever a CDPv1 packet is received by the storage system.

About this task

- The value of the `cdpd.holdtime` option applies to both nodes of an HA pair.
- The default value of hold time is 180 seconds.
- If an IP address is removed before the hold time expires, the CDP information is cached till the hold time expires.

Step

1. To configure the hold time, enter the following command:

   `options cdpd.holdtime` ***`holdtime`***

holdtime is the time interval, in seconds, for which the CDP advertisements are cached in the neighboring CDP-compliant devices. You can enter values ranging from 10 seconds to 255 seconds.

Setting the intervals for sending CDP advertisements

CDP advertisements are sent at periodic intervals. You can increase or decrease the intervals between the sending of each CDP advertisement, depending on the network traffic and change in the network topology. You can use the `cdpd.interval` option to configure the time interval for sending CDP advertisements.

About this task

The value of the `cdpd.interval` option applies to both the nodes of an HA pair.

Step

1. To configure the interval for sending CDP advertisements, enter the following command:

   options cdpd.interval *interval*

   interval is the time interval after which CDP advertisements should be sent. The default interval is 60 seconds. The time interval can be set between the range of 5 seconds and 900 seconds.

Viewing or clearing CDP statistics

You can analyze the CDP statistics to detect any network connectivity issues. You can use the `cdpd show-stats` command to view the CDP send and receive statistics. CDP statistics are cumulative from the time they were last cleared. To clear the CDP statistics, you can use the `cdpd zero-stats` command.

Before you begin

CDP must be enabled.

Step

1. Depending on whether you want to view or clear the CDP statistics, complete the following step:

If you want to...	Then, enter the following command...
View the CDP statistics	**cdpd show-stats**
Clear the CDP statistics	**cdpd zero-stats**

Example of showing the statistics before and after clearing them

The following example shows the CDP statistics before they are cleared:

```
system1> 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
```

This output displays the total packets that are received from the last time the statistics were cleared.

The following command clears the CDP statistics:

```
system1> cdpd zero-stats
```

The following output shows the statistics after they are cleared:

```
system1> cdpd show-stats

RECEIVE
 Packets:             0  | Csum Errors:         0  | Unsupported
Vers:    0
 Invalid length:      0  | Malformed:           0  | Mem alloc
fails:     0
 Missing TLVs:        0  | Cache overflow:      0  | Other
errors:        0


TRANSMIT
 Packets:             0  | Xmit fails:          0  | No
hostname:         0
 Packet truncated:    0  | Mem alloc fails:     0  | Other
errors:        0


OTHER
 Init failures:       0
```

After the statistics are cleared, the statistics get added from the time the next CDP advertisement is sent or received.

Viewing neighbor information by using CDP

You can view information about the neighboring devices connected to each port of your storage system, provided that the port is connected to a CDP-compliant device. You can use the `cdpd show-neighbors` command to view neighbor information.

Before you begin

CDP must be enabled.

About this task

If a network interface does not have an IP address configured, you can view the CDP information of the storage system from the switch. However, you cannot view the CDP information of the switch from the storage system.

Step

1. To view information about all CDP-compliant devices connected to your storage system, enter the following command:

   `cdpd show-neighbors`

Example

The following example shows the output of the `cdpd show-neighbors` command:

```
system1> 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 storage system. The "Remote Capability" column specifies the capabilities of the remote device that are connected to the network interface. The following capabilities are available:

- R—Router
- T—Transparent bridge
- B—Source-route bridge
- S—Switch
- H—Host
- I—IGMP
- r—Repeater

- P—Phone

How to monitor your storage system with SNMP

You can configure SNMP to monitor your storage system to avoid issues before they occur and to respond to issues when they occur. Managing SNMP involves configuring SNMP users and configuring SNMP traphosts for specific events.

SNMP is enabled by default on your storage system. SNMP network management workstations or managers can query your storage system's 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 storage system has read-only privileges—that is, it cannot be used for any set operations or for taking a corrective action in response to a trap. Data ONTAP provides an SNMP agent compatible with SNMP versions v1, v2c, and v3. SNMPv3 offers advanced security by using passphrases and encryption.

Types of SNMP traps

SNMP traps capture system monitoring information that is sent as an asynchronous notification from the SNMP agent (storage system) to the SNMP manager. There are three types of SNMP traps: standard, built-in, and user-defined.

A trap can be used to periodically check for different operational thresholds or failures, which are defined in the MIB. If a threshold is reached or failure is detected, the SNMP agent on the storage system sends a message (trap) to the traphosts alerting them of the event.

Standard SNMP traps	These traps are defined in RFC 1215. There are five standard SNMP traps that are supported by Data ONTAP: coldStart, warmStart, linkDown, linkUp, and authenticationFailure.
Built-in SNMP traps	Built-in traps are predefined in Data ONTAP and are automatically sent to the network management stations on the traphost list if an event occurs. These traps are defined in the custom MIB, such as diskFailedShutdown, cpuTooBusy, and volumeNearlyFull. Each built-in trap is identified by a unique trap code.
User-defined SNMP traps	User-defined traps are defined by the `snmp traps` command. These traps are sent using proxy trap IDs 11 through 18, which correspond to a trap's MIB priority.

What MIBs are

A MIB file 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 (OID). Each OID identifies a variable that can be read by using SNMP.

As MIBs are not configuration files and Data ONTAP does not read these files, SNMP functionality is not affected by MIBs. Data ONTAP provides two MIB files:

- A custom MIB (`/etc/mib/netapp.mib`)
- An Internet SCSI (iSCSI) MIB (`/etc/mib/iscsirfc4544.mib`)

SNMPv3 supports the MIB-II specification and the MIBs of your storage system. The following MIB-II groups are supported:

- System
- Interfaces
- Address translation
- IP
- ICMP
- TCP
- UDP
- SNMP

Data 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.

Data ONTAP also provides a short cross-reference between object identifiers (OIDs) and object short names in the `/etc/mib/traps.dat` file. This file is useful for creating user-defined traps.

Note: The latest versions of the Data ONTAP MIBs and `traps.dat` files are available online on the NetApp Support Site. However, the versions of these files on the web site do not necessarily correspond to the SNMP capabilities of your Data ONTAP version. These files are provided to help you evaluate SNMP features in the latest Data ONTAP version.

Related information

NetApp Support Site: support.netapp.com

What the SNMP agent does

The storage system includes an SNMP agent that responds to queries and sends traps to network management stations.

The SNMP agent on the storage system has read-only privileges—that is, it cannot be used to take corrective action in response to a trap.

Note: Starting with Data ONTAP 7.3.1, the SNMP agent supports IPv6 transport.

How to configure the SNMP agent

You need to configure the SNMP agent on your storage system to set SNMP values and parameters.

To configure the SNMP agent on your storage system, you need to perform the following tasks:

- Verify that SNMP is enabled.

  Note: SNMP is enabled by default in Data ONTAP.
- If you are running SNMPv3, configure SNMPv3 for read-only access.
- Enable traps.
  Although SNMP is enabled by default, traps are disabled by default.
- Specify host names of one or more network management stations.
  Traps can only be sent when at least one SNMP management station is specified as a traphost. Trap notifications can be sent to a maximum of eight network management stations.

  Note: The SNMP agent can send traps over IPv6 transport to the traphosts whose IPv6 address is configured on the storage system. You can specify traphosts by their IPv6 addresses, but not by their host names.

You can perform the following tasks after configuring SNMP:

- Provide courtesy information about storage system location and contact personnel.
- Specify SNMP communities.
  Community strings function as group names to establish trust between SNMP managers and clients. Data ONTAP supports only read-only communities.

  Note: No more than eight communities are allowed.

Note: Storage systems in an HA configuration can have different SNMP configurations.

Related concepts

User-defined SNMP traps on page 136

Enabling or disabling SNMP

You can enable or disable SNMP from the command-line interface by entering the `options snmp.enable` command.

Step

1. To enable or disable SNMP, enter the following command:

   `options snmp.enable {on|off}`

   `on`—Enables SNMP

   `off`—Disables SNMP

Configuring SNMPv3 users

To access MIB objects by using SNMPv3, you should create users with `login-snmp` capability.

Steps

1. Enter the following command to create a role with `login-snmp` capability:

   `useradmin role add *role_name* -a login-snmp`

   `role_name` is the role name with `login-snmp` capability.

   Example

   `useradmin role add myrole1 -a login-snmp`

2. Enter the following command to create a group and add the created role to that group:

   `useradmin group add *group_name* -r *role_name*`

   `group_name` is the group name to which you want to add the created role, *`role_name`*.

   Example

   `useradmin group add mygroup1 -r myrole1`

3. Enter the following command to create a user and add the user to the created group:

   `useradmin user add *user_name* -g *group_name*`

   `user_name` is the user name belonging to the group, *`group_name`*.

   Example

   `useradmin user add myuser1 -g mygroup1`

4. Create a password for the new user.

   Ensure that the password has a minimum of eight and a maximum of sixteen characters.

5. Enter the `snmpwalk` command through the system MIB:

snmpwalk -v 3 -u *user_name* -l authNoPriv -A *password storage_system* system

password is the user's password that you entered in Step 3.

storage_system is the storage system that contains the MIBs.

Example

```
snmpwalk -v 3 -u myuser1 -l authNoPriv -A johndoe123 host1 system
```

Setting SNMP access privileges

You can set SNMP access privileges on a host or an interface by using the command-line interface. The `snmp.access` option defines a method to restrict SNMP access to the storage system.

Step

1. Enter the following command:

   options snmp.access *access_spec*

   access_spec consists of keywords and their values.

   The syntax for *access_spec* is described in the na_protocolaccess(8) man page. For more information about the `options` command, see the na_options(1) man page.

   Example
   To allow access to SNMP for network interfaces e0, e1, and e2, enter the following command:

   ```
   options snmp.access if=e0,e1,e2
   ```

Related tasks

Restricting protocol access on page 37

Viewing or modifying your SNMP configuration

You can use the `snmp` command to view or modify your SNMP configuration values.

Step

1. Enter the following command:

   snmp {*options values*}

   options are the available options for the `snmp` command, such as `authtrap`, `community`, `contact`, `init`, `location`, `traphost`, and `traps`.

   values are the values that you want to set for a particular option.

   If you specify one or more values for an option of the SNMP commands, the value of that option is set or changed. However, if no values are specified, the current value of that option is returned.

Related references

SNMP command syntax on page 132

SNMP command syntax

If you specify one or more values for an option of the SNMP commands, the value of that option is set or changed. However, if no values are specified, the current value of that option is returned.

The following table describes the syntax and parameters of SNMP commands.

Command	Description
`snmp`	Displays the current values of all SNMP options, such as `init`, `community`, `contact`, and `traphost`.
`snmp authtrap [0|1]`	**With a value**: Enables (with value 1) or disables (with value 0) authentication failure traps on the SNMP agent. **Without a value**: Displays the current value of `authtrap` set in Data ONTAP.
`snmp community`	Displays the current list of communities.
`snmp community add ro`*`community`*	Adds a community. **Default value**: The default community for the SNMP agent in Data ONTAP is public. The only access mode available on storage systems is the default `ro` (read-only).
`snmp community delete {all | ro`*`community`*` }`	Deletes one or all communities.
`snmp contact [`*`contact`*`]`	**With a value**: Sets the contact name for your storage system. You must enclose the contact string in single quotes (' ') if the string contains spaces. You can enter a maximum of 255 characters for the contact information. **Without a value**: Displays the current contact name set in Data ONTAP.
`snmp init [0|1]`	**With a value**: Enables (with value 1) or disables (with value 0) built-in traps and the traps defined using the `snmp traps` command. **Without a value**: Displays the current value of `snmp init` in Data ONTAP. **Default value**: By default, SNMP traps are disabled in Data ONTAP; the system uses the equivalent of `snmp init 0`.

Command	Description
`snmp location [location]`	**With the option**: Sets the location of your storage system. You must enclose the *location* string in single quotes (' ') if the string contains spaces. **Without the option**: Displays the current location set in Data ONTAP.
`snmp traphost [{add|delete} { hostname| ipaddress}]`	**With the option**: Adds or deletes SNMP hosts that receive traps from Data ONTAP. When IPv6 is enabled on your storage system, IPv6 traphosts can be added and deleted. You can specify IPv6 addresses, and not host names, to identify IPv6 traphosts. **Without the option**: Displays the current traphosts set in Data ONTAP.
`snmp traps [options]`	Displays the list of user-defined traps set in Data ONTAP

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.

If the security level is set to authNoPriv, authentication is performed by using the user's authKey to sign the message being sent. The authProtocol parameter must be MD5. The authKey parameters are generated from a passphrase that must have a minimum of eight characters. If the security level is set to authNoPriv, you must enter the following parameters:

Parameter	Command-line option	Description
securityName	-u *Name*	User name must not exceed 31 characters.
authProtocol	-a (MD5)	Authentication type must be MD5. **Note:** Data ONTAP does not support SHA authentication protocol.
authKey	-A *PASSPHRASE*	Passphrase with a minimum of eight characters.
securityLevel	-l (authNoPriv)	Security level: must be Authentication, No Privacy. **Note:** Data ONTAP does not support retrieving MIB values using the AuthPriv and noAuthNoPriv security levels.

Parameter	Command-line option	Description
context	-n *CONTEXTNAME*	Sets the context name used for SNMPv3 messages.

Example: SNMP commands

You can use the `snmpget`, `snmpwalk`, `snmpbulkget`, and `snmpbulkwalk` commands to retrieve information from network elements with SNMP agents.

snmpwalk

The following command retrieves all the variables under the system sys1:

```
snmpwalk -Os -c public -v 1 sys1 system
sysDescr.0 = STRING: Data ONTAP Release 7.3.1
sysObjectID.0 = OID: enterprises.789.2.3
sysUpTimeInstance = Timeticks: (121596665) 14 days, 1:46:06.65
sysContact.0 = STRING:
sysName.0 = STRING: sys1.lab.example.com
sysLocation.0 = STRING:
sysServices.0 = INTEGER: 72
```

The following command is an example of an SNMP request from an IPv6 client:

```
snmpwalk -v2c -c public udp6:[2001:0db8:85a3:0:0:8a2e:0370:99]:161 system
SNMPv2-MIB::sysDescr.0 = STRING: Data ONTAP Release 7.3.1
SNMPv2-MIB::sysObjectID.0 = OID:
SNMPv2-SMI::enterprises.789.2.3
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (11415057) 1 day,7:42:30.57
SNMPv2-MIB::sysContact.0 = STRING:
SNMPv2-MIB::sysName.0 = STRING:n3700-183-85.sys1.lab.example.com
SNMPv2-MIB::sysLocation.0 = STRING:
SNMPv2-MIB::sysServices.0 = INTEGER: 72
```

The following command is an example of an SNMPv3 request to retrieve all variables under the system sys1:

```
snmpwalk -v 3 -u joe -l authNoPriv -A joe12 sys1 system
SNMPv2-MIB::sysDescr.0 = STRING: Data ONTAP Release 7.3.1
SNMPv2-MIB::sysObjectID.0 = OID:
SNMPv2-SMI::enterprises.789.2.3
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (121622059) 14
days, 1:50:20.59
SNMPv2-MIB::sysContact.0 = STRING:
SNMPv2-MIB::sysName.0 = STRING: sys1.lab.example.com
SNMPv2-MIB::sysLocation.0 = STRING:
SNMPv2-MIB::sysServices.0 = INTEGER: 72
```

Note: You need to enter authentication information for using SNMPv3.

snmpget

The following command retrieves the `system.sysDescr.0` object from the host sys1 by using the `public` community string:

```
snmpget -c public sys1 system.sysDescr.0
system.sysDescr.0 = Data ONTAP Release 8.0  Mon Mar 16 16:56:43 IST 2010
```

The following command retrieves the value of an ICMP object (OID=56.1.1.1.1) from the host sys1:

```
snmpget -c public -v 2c sys1 .1.3.6.1.2.1.56.1.1.1.1
    56.1.1.1.1 = Counter32: 0
```

snmpbulkget

The following command retrieves the system object `sysDescr.0` and the first three objects in the ifTable:

```
snmpbulkget -v2c -Cn1 -Cr3 -Os -c public sys1 system ifTable
sysDescr.0 = STRING: Data ONTAP Release 7.3.1
ifIndex.1 = INTEGER: 1
ifIndex.2 = INTEGER: 2
ifDescr.1 = STRING: "lo0"
```

The following example shows a part of the output from retrieving all variables under the IPv6 object (OID=55.1):

```
snmpbulkget -c public -v 2c 192.0.2.19 .1.3.6.1.2.1.55.1
    55.1.1.0 = 2
    55.1.2.0 = 64
    55.1.3.0 = Gauge32: 4
    55.1.4.0 = Counter32: 0
    55.1.5.1.1.1 = 1
    55.1.5.1.2.1 = "ns0"
    55.1.5.1.3.1 = OID: .ccitt.zeroDotZero
    55.1.5.1.4.1 = 1500
    55.1.5.1.5.1 = 65535
    55.1.5.1.6.1 = IpAddress: 00 00 00 00 00 00 00 00 02 05 00 FF FE 00 02 AB
    55.1.5.1.7.1 = 64
    55.1.5.1.8.1 =  Hex: 00 05 00 00 02 AB
    55.1.5.1.9.1 = 1
    55.1.5.1.10.1 = 1
```

snmpbulkwalk

The following command retrieves all the variables under the system sys1:

```
snmpbulkwalk -v2c -Os -c public sys1 system
sysDescr.0 = STRING: Data ONTAP Release 7.3.1
sysObjectID.0 = OID: enterprises.789.2.3
sysUpTimeInstance = Timeticks: (121603434) 14 days, 1:47:14.34
sysContact.0 = STRING:
sysName.0 = STRING: sys1.lab.example.com
sysLocation.0 = STRING:
sysServices.0 = INTEGER: 72
```

The following example shows a part of the output from retrieving all the variables for the UDP object:

```
snmpbulkwalk -c public -v 2c 192.0.2.19 udp
    udp.udpInDatagrams.0 = Counter32: 347
    udp.udpNoPorts.0 = Counter32: 4
    udp.udpInErrors.0 = Counter32: 0
    udp.udpOutDatagrams.0 = Counter32: 138
```

```
    udp.udpTable.udpEntry.udpLocalAddress.0.0.0.0.69 = IpAddress: 00 00 00 00
    udp.udpTable.udpEntry.udpLocalAddress.0.0.0.0.111 = IpAddress: 00 00 00 00
```

User-defined SNMP traps

If the predefined built-in traps are not sufficient to create alerts for conditions you want to monitor, you can create user-defined traps in Data ONTAP.

Before you define a new trap, you should consult the Data ONTAP MIBs to see if any existing traps serve your purpose.

How SNMP traps work

You can set SNMP traps to inspect the value of MIB variables periodically. Whenever the value of a MIB variable meets the conditions you specify, a trap is sent to the network management stations on the traphost list. The traphost list specifies the network management stations that receive the trap information.

You can set traps on any numeric variable in the MIB. For example, you can set a trap to monitor the fans on your storage system. You can then set the SNMP application on your network management station to display a flashing message on your console when a fan stops working.

Traps are persistent across reboots until you remove it or modify it.

Follow these guidelines when creating traps:

- Use the `/etc/mib/traps.dat` file to find Object Identifiers (OIDs) for objects in the MIB files of your storage system.
- Ensure that the trap can be generated in the storage system environment.
- Do not set traps on tabular data.

  It is possible to set traps on row entries in a sequence—for example, an entry in a table. However, if the order in the table is changed by adding or removing rows, you will no longer be trapping the same numeric variables.

How to define or modify a trap

You can define traps or modify traps you have already defined by entering values from the command-line interface or in a configuration file.

You must supply the following elements when you create or modify traps:

- Trap name

  Trap name is the name of the user-defined trap you want to create or change. A trap name must not have any embedded periods.
- Trap parameters
- Parameter values

Note: When you create a user-defined trap, it is initially disabled by default. You must enable a trap before it can be triggered. You enable traps by using the `snmp traps` command.

Viewing or modifying trap values from the command-line interface

You can view or modify your trap values by using the `snmp traps` command.

Step

1. Enter the following command:

   `snmp traps {`*`options variables`*`}`

   `options` are the options for SNMP traps such as `walk`, `load`, `trapname`, and so on.

   `variables` are values for the specified option.

Example: Trap definitions

You can define a group of traps by using the command-line interface.

The following example sets a group of traps. The trap descriptions are numbered in brackets.

`snmp traps cifstotalops.var snmp.1.3.6.1.4.1.789.1.7.3.1.1.1.0`
[1]

`snmp traps cifstotalops.trigger level-trigger`

`snmp traps cifstotalops.edge-1 1000000`
[4]

`snmp traps cifstotalops.interval 10`
[2]

`snmp traps cifstotalops.backoff-calculator step-backoff`
[5]

`snmp traps cifstotalops.backoff-step 3590`
[5]

`snmp traps cifstotalops.rate-interval 3600`
[3]

`snmp traps cifstotalops.priority alert`

`snmp traps cifstotalops.message snmp.1.3.6.1.4.1.789.1.7.3.1.1.1.0`

A cifstotalops trap [1] is evaluated every 10 seconds [2]. The value received from the previous evaluation and the current value are used to calculate the number of CIFS operations per hour [3]. If the number exceeds one million [4], the trap triggers and continues to be triggered every hour [5] until the total number of CIFS operations drops below one million.

Command syntax for SNMP trap parameters

If you specify one or more values for an option of the SNMP commands, the value of that option is set or changed. However, if no values are specified, the current value of that option is returned.

The following table describes the syntax and parameters for the `snmp traps` command.

Command	Description
`snmp traps`	Displays the list of user-defined traps set in Data ONTAP.
`snmp traps [enable|disable|reset|delete]` *`trapname`*	Enables, disables, resets, or deletes the trap *`trapname`*. If you do not specify a trap name, all traps defined so far are acted on.
`snmp traps walk` *`prefix`*	Walks (traverses in order) the trap list by prefix; that is, lists all traps that have names beginning with *`prefix`*.
`snmp traps load` *`trap_list_filename`*	Loads a set of traps from a configuration file. The file contains a list of traps and parameters without the `snmp traps` command preceding each trap. If the specified file name is defaults, traps are read from the `/etc/defaults/` traps file.
`snmp traps` *`trapname.parm value`*	Defines or changes a user-defined trap parameter.

SNMP trap parameters

You must specify certain parameters to create SNMP traps.

You can specify the following SNMP trap parameters with the `snmp traps` command:

- `var`
- `trigger`
- `edge-1`
- `edge-2`
- `edge-1-direction`
- `edge-2-direction`
- `interval`
- `interval-offset`
- `rate-interval`
- `backoff-calculator`
- `backoff-step`
- `backoff-multiplier`
- `priority`

- `message`

The var parameter

The `var` parameter associates a user-defined trap name (specified by the *`trapname`* variable in the `snmp traps` command) with a specific MIB object. The MIB object is specified in the value field of the `snmp traps` command. It must be in the format `snmp.`*`oid`*, where *`oid`* is an Object Identifier (OID).

The `traps.dat` file, which is located in the `/etc/mib` directory on your storage system, can help you determine OIDs. This file maps MIB objects' short names in the Data ONTAP MIB files to their numeric OIDs. For more information about a particular OID, see the MIB.

The trigger parameter

The trigger parameter specifies the type of triggers that you can set for a trap. If a trap is triggered, data about the event that caused the trigger is sent to the network management stations.

You can specify the following values for the `trigger` parameter:

single-edge-trigger	Triggers a trap and sends data when the value of the trap's MIB variable crosses an edge (a value that you specify) for the first time.
double-edge-trigger	Triggers a trap and sends data when either of two edges is crossed. A double-edge-trigger enables you to set two edges, each with its own direction.
level-trigger	Triggers a trap and sends data whenever the trap's value crosses a specified edge value.
change-trigger	Keeps track of the last value received from the trap. If the current value differs from the previously received value, the trap is triggered.
always-trigger	Enables a trap to always trigger at the specified evaluation interval (specified by the `interval` parameter). For example, a trap can trigger every 24 hours for the agent to send the total number of CIFS operations to an SNMP manager.

The edge-1 and edge-2 parameters

The edge-1 and edge-2 parameters of a trap specify the threshold values that are compared during trap evaluation to determine whether to fire a trap and send data.

The edge-1 parameter specifies the value for the edge in a single-edge-triggered trap or the first edge in a double-edge-triggered trap. The default value for the edge-1 parameter is MAXINT.

The edge-2 parameter specifies the value for the second edge in a double-edge-triggered trap. The default value for the edge-2 parameter is 0.

The edge-1-direction and edge-2-direction parameters

The edge-1-direction and edge-2-direction parameters enable you to set or change the direction that is used to evaluate a trap. The edge-triggered traps only send data when the edge is crossed in either the up or down direction.

The default value for the edge-1-direction parameter is `up` and for the edge-2-direction parameter is `down`.

Note: You enter the direction values on the same line as the edge value when you run the `snmp traps` command.

The interval parameter

The `interval` parameter is the time, in seconds, between evaluations of a trap.

A trap can only send data as often as it is evaluated, even if the edge values are exceeded sooner. The default value for the `interval` parameter is 3600.

Note: The maximum value that can be specified for the `interval` parameter in Data ONTAP is 2147482 seconds, which is approximately equal to 24 days.

The interval-offset parameter

The `interval-offset` parameter is the amount of time, in seconds, until the first trap evaluation.

The default value for the `interval-offset` parameter is 0. You can set it to a nonzero value to prevent too many traps from being evaluated at once (for example, at system startup).

The rate-interval parameter

The `rate-interval` parameter specifies the time, in seconds, in which the change in value of a trap's variable (rate of change) is expressed.

If the `rate-interval` value is set for a trap, the samples of data obtained at the interval points (set using the `interval` parameter) for a trap variable are used to calculate the rate of change. If the calculated value exceeds the value set for the `edge-1` or `edge-2` parameter, the trap is fired.

For example, to obtain the number of CIFS operations per hour, you specify a `rate-interval` of 3600. If `rate-interval` is set to 0, no sampling at interval points occurs and trap evaluation proceeds as with any other kind of trap. The default value for the `rate-interval` parameter is 0.

The backoff-calculator parameter

The `backoff-calculator` parameter enables you to change the trap evaluation interval for a trap after a trap fires.

After a trap fires and sends data, you might not want it to be evaluated so often. For instance, you might want to know within a minute of when a file system is full, but only want to be notified every hour thereafter that it is still full.

The `backoff-calculator` parameter can take the following values in the value variable field:

- `step-backoff`
- `exponential-backoff`
- `no-backoff`

The default value for the `backoff-calculator` parameter is `no-backoff`.

The backoff-step parameter

The `backoff-step` parameter specifies the number of seconds by which the trap evaluation interval is increased.

If a trap interval is 10 and its backoff-step is 3590, the trap is evaluated every 10 seconds until it fires the first time and sends data, and once an hour thereafter. The default value for the `backoff-step` parameter is 0.

The backoff-multiplier parameter

The `backoff-multiplier` parameter specifies the value by which to multiply a trap's evaluation interval each time it fires.

If you set `backoff-calculator` to `exponential-backoff` and `backoff-multiplier` to `2`, the interval doubles each time the trap fires. The default value of the `backoff-multiplier` parameter is 1.

The priority parameter

The priority parameter sets the priority of a trap. If several traps are scheduled to be triggered at the same time, you can use the priority parameter to decide which trap is serviced first.

The possible values for the priority parameter, from highest to lowest, are as follows:

- `emergency`
- `alert`
- `critical`
- `error`
- `warning`
- `notification`
- `informational`
- `debug`

The default value for the priority parameter is `notification`.

The message parameter

The `message` parameter specifies a message that goes out with a trap.

The message can be a string of text or simply the SNMP OID, in the form `snmp.oid`. If you specify the OID as your message, Data ONTAP sends the information that was trapped concerning the OID.

If you do not specify a message parameter for a trap, when the trap fires you see a string with the numerical OID value and its priority level.

For example, the following string is sent to the network management stations for the trap `cpuUpTime` if the message parameter is not set:

```
cpuUpTime == 10562288.priority == notification
```

Note: If the message is a string that includes spaces, you must enclose the string in quotation marks (" ").

How to diagnose network problems

You can diagnose problems on your network by using commands such as `cdpd`, `netdiag`, `ping`, and `pktt`. You can also use commands such as `ping6`, `ndp`, and `traceroute6` to diagnose IPv6 problems.

`cdpd`

You can use the `cdpd` command to view information about the devices that advertise themselves by using the CDPv1 protocol. You can also view information about the CDP neighbors of the storage system and thus detect network connectivity.

`netdiag`

You can use the `netdiag` command to continuously gather and analyze statistics, and perform diagnostic tests. These diagnostic tests identify and report problems with your physical network or transport layers and suggest remedial action.

For more information about the `netdiag` command along with all available options, see the na_netdiag(1) man page.

`ping`

You can use the `ping` command to test whether your storage system can reach other hosts on your network.

For more information about the `ping` command, see the na_ping(1) man page.

`pktt`

You can use the `pktt` command to trace the packets sent and received in the storage system's network.

For more information about the `pktt` command, see the na_pktt(1) man page.

`ping6`

You can use the `ping6` command to reach IPv6 hosts. Starting with Data ONTAP 7.3.3, you can use the `ping` command for reaching IPv6 hosts. You can use the `ping6` and `ping` commands with all types of IPv6 addresses.

The `-d` option in the `ping6` command specifies the number of data bytes to be sent. In addition, you might need to specify the `-b` option to extend the socket buffer size.

You must use the `-b` option with the `ping6` command when pinging hosts with jumbo frames. For the pinging to succeed with jumbo frames, the buffer must be large enough to reassemble IP fragments.

For example, when pinging an IPv6 address with an 8900 byte payload and specifying a 9000 byte buffer, you should use the following command:

```
ping6 -d 8900 -b 9000 2001:0db8::99
```

In the previous example, setting the buffer size to 8901 or 8902 bytes might not be adequate and might cause the `ping6` command to fail. Increasing the buffer size to 10000 allows the ping to succeed in both directions.

ndp You can use the `ndp` command to control the address mapping table used by Neighbor Discovery Protocol (NDP).

For more information about the `ndp` command, see the na_ndp(1) man page.

traceroute6 You can use the `traceroute6` command to trace the route that the IPv6 packets take to a network node.

For more information about the `traceroute6` command, see the na_traceroute6(1) man page.

Related references

Error codes for the netdiag command on page 186

Diagnosing transport layer problems

You can use the `netdiag -t` command to diagnose problems with the transport layer in your storage system.

Step

1. Enter the following command:

   `netdiag -t`

Example

A storage system whose TCP window size is smaller than the recommended value displays the following output:

```
Performing transport layer diagnostics.....
The TCP receive window advertised by CIFS client
192.0.2.13 is 8760. This is less than the recommended
value of 32768 bytes. You should increase the TCP receive
buffer size for CIFS on the client. Press enter to continue.
```

Viewing diagnostic results

You can use the `netdiag -s` command to view a summary of the various diagnostic checks and tests performed on your storage system.

About this task

If you enable the IPv6 option, you can view the IPv4 and IPv6 statistics in the network layer diagnostic summary.

Step

1. Enter the following command:

   `netdiag -s`

Example

The following output shows some issues in IPv6 configuration of the network layer.

```
netdiag -s

Physical Layer Diagnostics Summary:

Interface  H/W   Link  Configured   Speed     Duplex     AutoNeg
          Status           UP      Mismatch  Mismatch   Mismatch
e0a          OK   -         N          -         -          -
e0b          OK   -         N          -         -          -
e0c          OK   Y         Y          N         -          -
e0d          OK   -         N          -         -          -


Network Layer Diagnostics Summary:

Protocol         Status
IP                 OK
IPv6              Prob


Transport Layer Diagnostics Summary:

Protocol         Status
TCP                OK
UDP                OK

Use netdiag without the -s option for details
```

How to diagnose ping problems

You can use the Data ONTAP ping throttling mechanism and its `ip.ping_throttle.drop_level` option to help avoid denial-of-service attacks that can occur when using ICMP.

The ping throttling mechanism is active in intervals of 1 second. If the number of ICMP echo and reply packets that the storage system receives in a one-second interval exceeds the ping throttling threshold, the storage system drops all subsequent packets that are received within that one-second interval.

Note: Regardless of whether the ping throttling threshold has been reached, clients that send more than 16 packets per second to a storage system might experience packet loss. To allow clients to send more than 16 packets per second, you must disable ping throttling.

If your storage system supports a very large number of CIFS clients that use ICMP pings to determine CIFS shares accessibility, you can increase the ping throttling threshold value in the `ip.ping_throttle.drop_level` option.

If a large number of CIFS clients are experiencing temporary or persistent unavailability of the storage system, you should check to see if the ping throttling threshold has been exceeded for the storage system. If so, you can increase the ping throttling threshold value.

Increasing the ping throttling threshold value

If your storage system supports a very large number of CIFS clients that use ICMP pings to determine CIFS shares accessibility, you might need to increase the ping throttling threshold value.

Step

1. Enter the following command:

   `options ip.ping_throttle.drop_level` ***`packets_per_second`***

   `packets_per_second` specifies the maximum number of ICMP echo or echo reply packets (ping packets) that the storage system accepts per second. Any further packets within 1 second are dropped. The default value is 150.

Checking the ping throttling threshold status

If a large number of CIFS clients are experiencing temporary or persistent unavailability of the storage system, you should check if the ping throttling threshold has been exceeded for the storage system.

Step

1. Enter the following command:

   `netstat -p icmp`

Result

The resulting report lists the number of pings and ping replies that have been dropped, if any.

If the number of pings dropped, the number of ping replies dropped, or the number of both pings and ping replies dropped is greater than zero, you should increase the `ip.ping_throttle.drop_value` option to a number that is higher than the current value.

Disabling ping throttling

To allow clients to send more than 16 packets per second, you need to disable ping throttling.

Step

1. Enter the following command:

```
options ip.ping_throttle.drop_level 0
```

Protecting your storage system from forged ICMP redirect attacks

You can disable ICMP redirect messages to protect your storage system against forged ICMP redirect attacks.

About this task

To efficiently route a series of datagrams to the same destination, your storage system maintains a route cache of mappings to next-hop gateways. If a gateway is not the best next-hop for a datagram with a specific destination, the gateway forwards the datagram to the best next-hop gateway and sends an ICMP redirect message to the storage system. By forging ICMP redirect messages, an attacker can modify the route cache on your storage system, causing it to send all of its communications through the attacker. The attacker can then hijack a session at the network level, easily monitoring, modifying, and injecting data into the session.

Step

1. Enter the following command:

   ```
   options ip.icmp_ignore_redirect.enable on
   ```

   Your storage system now ignores ICMP redirect messages.

   For more information about the `ip.icmp_ignore_redirect.enable` option, see the na_options(1) man page.

   Note: By default, the `ip.icmp_ignore_redirect.enable` option is `off`.

Improving TCP network congestion with Appropriate Byte Counting

Appropriate byte counting (ABC) is an algorithm implemented in the TCP stack of Data ONTAP that enables you to improve TCP performance and security. Starting from Data ONTAP 8.1, ABC is enabled by default on all the storage controllers.

A TCP sender maintains an estimate of the number of bytes it can send into the network in a state variable called the congestion window (cwnd). A TCP receiver sends acknowledgements (ACKs) for every data packet received. When the TCP receiver uses delayed ACKs, it might acknowledge every other data packet received. The TCP congestion control algorithms of Slow Start and Congestion Avoidance, as described in RFC 2581, control the update to cwnd. The updates are based on the incoming TCP ACKs. ABC is an improvement over these algorithms as it uses the actual bytes acknowledged in the incoming TCP ACKs to update cwnd instead of just using the incoming ACK as a trigger to update the cwnd. This leads to a more accurate cwnd estimate and improvement in TCP performance.

A misbehaving TCP receiver or an attacker might send ACKs that acknowledge very few bytes. This can cause a TCP sender that uses just the number of ACKs to increase cwnd to aggressively send large amounts of data into the network. ABC ensures that the TCP sender rate matches that of the exact number of bytes acknowledged and prevents the attack on the CPU of the sender, thereby enhancing the TCP security.

Increasing the upper limit of initial window in TCP

In TCP connections, the upper limit of the initial window is usually set between one and two segments. The implementation of RFC 3390 helps improve the transmission time in slow TCP connections.

About this task

- You can increase the upper limit of the initial window from one or two segments to two and four segments.
- In an HA pair, you must set the same value in both the nodes.

Step

1. To increase the upper limit of the initial window, enter the following command:

   `options ip.tcp.rfc3390.enable{on|off}`

   `on`—Enables the implementation.

   `off`—Disables the implementation.

Specifying the limit for increasing the congestion window in ABC

In TCP connections implementing ABC, you can specify the rate of increase of the congestion window during the slow start phase. This limit enables you to improve TCP performance.

About this task

- You can increase the congestion window's limit only when ABC is used in TCP congestion control.
- In an HA pair, you must set the same value in both the nodes.
- The default value of this option is set to 2.
- If the rate at which the packets are dropped is increased, probably due to network congestion, then you should set the limit value to 1.

Step

1. To specify the limit for increasing the congestion window, enter the following command:

   `options ip.tcp.abc.l_limit {1|2}`

   `1`—This is the value for a conservative increase of the congestion window.

Network interface statistics

You can use the `ifstat` command to view statistics for the network interfaces supported by Data ONTAP. To determine the Ethernet controllers in your system, you can use the `sysconfig` command.

Statistics for Gigabit Ethernet controller VI, VII, and G20 interfaces

The `ifstat` command output displays several statistics when you use the command for the Gigabit Ethernet controllers and G20 interfaces.

The statistics in this section are for the following controllers:

- 10/100/1000 Ethernet controller VI and VII
- Gigabit Ethernet controller VI
- 10/100/1000 Ethernet controller G20
- Gigabit Ethernet controller G20

RECEIVE section statistics

The following table describes the statistics in the RECEIVE section of the `ifstat` command output.

Statistic	Definition
`Frames/second`	Rate of received frames per second.
`Bytes/second`	Rate of received bytes per second.
`Errors/minute`	Rate of errors (which led to frames being lost) per minute.
`Discards/minute`	Rate per minute of packets discarded due to unavailable resources.
`Total frames`	Total frames that are received on the interface.
`Total bytes`	Total bytes that are received on the interface.
`Total errors`	Total errors that occur on the interface.
`Total discards`	Total number of packets that were discarded even though no errors were detected. This number is a sum of the "No buffers", "Bus overruns", and "Queue overflows" statistics.
`Multi/broadcast`	Total number of multicast or broadcast packets received.

Statistic	Definition
`Alignment errors`	Number of frames that are both misaligned and contain CRC errors.
`Non-primary u/c`	Number of Ethernet frames received for the partner's MAC address after a failover in an HA pair.
`Tag drop`	Number of tagged frames dropped on an interface that is not configured to support VLAN tagging.
`Vlan tag drop`	Number of tagged frames dropped that do not match the VLAN tags configured on the interface.
`Vlan untag drop`	Number of untagged frames dropped on an interface that is configured to be part of a VLAN.
`CRC errors`	Number of packets received with bad CRC.
`Bad length`	Total number of received packets with a bad length. These are frames counted as undersize, fragment, oversize, or jabber.
`Runt frames`	Number of received frames that were less than the minimum size (64 bytes) and had a valid CRC.
`Fragment`	Number of received frames that were less than the minimum size and had a bad CRC.
`Long frames`	Number of received frames that were greater than the maximum size and had a valid CRC.
`Jabber`	Number of received frames that were greater than the maximum size and had a bad CRC.
`Bus overruns`	Number of times the adapter's receive FIFO overflowed and a packet was dropped. This occurs when the bus is very busy and the adapter cannot transfer data into host memory. This might also occur when your storage system CPU is very busy and cannot process the received packets fast enough.
`Queue overflows`	Number of frames dropped on receive due to the driver receive queue overflowing.
`No buffer`	Number of times the driver could not allocate a buffer and a packet was dropped. This might happen when your storage system is very busy. If the count increases continually, it might indicate that a software component is not returning buffers.
`Xon`	Number of XON frames received when receive or full flow control is enabled.

Statistic	Definition
`Xoff`	Number of XOFF frames received when receive or full flow control is enabled.
`Jumbo`	Number of good packets received that were larger than the standard Ethernet packet size when jumbo frames are enabled.
`Reset`	Number of times the driver reset the NIC because the NIC was in a bad state.
`Reset1`	Number of times the driver reset the NIC because the NIC was in a bad state.
`Reset2`	Number of times the driver reset the NIC because the NIC was in a bad state.

TRANSMIT section statistics

The following table describes the statistics in the TRANSMIT section of the `ifstat` command output.

Statistic	Meaning
`Frames/second`	Rate of transmitted frames per second.
`Bytes/second`	Rate of transmitted bytes per second.
`Errors/minute`	Rate of errors (which led to frames being lost) per minute.
`Discards/minute`	Rate per minute of packets discarded due to unavailable resources.
`Total frames`	Total frames that are transmitted on the interface.
`Total bytes`	Total bytes that are transmitted on the interface.
`Total errors`	Total errors that occur on the interface.
`Total discards`	Total number of packets that were discarded even though no errors were detected. This number is a sum of the "No buffers" and "Queue overflows" statistics.
`Multi/broadcast`	Total number of multicast or broadcast packets transmitted.
`No buffers`	Number of times the driver failed to allocate a buffer for the transmit packet.
`Queue overflow`	Number of outgoing packets dropped because the driver's queue was full. It might indicate a system problem.

Statistic	Meaning
`Max collisions`	Number of frames that were not transmitted because they encountered the maximum number of allowed collisions. Only valid in half-duplex mode.
`Single collision`	Number of frames that encountered exactly one collision. Only valid in half-duplex mode.
`Multi collisions`	Number of frames that encountered more than one collision, but less than the maximum allowed. Only valid in half-duplex mode.
`Late collisions`	Number of collisions that occurred outside the collision window. Only valid in half-duplex mode.
`Xon`	Number of XON frames transmitted when send or full flow control is enabled.
`Xoff`	Number of XOFF frames transmitted when send or full flow control is enabled.
`Timeout`	Number of times the adapter's transmitter hung and the adapter had to be reset. This can happen when the cable is pulled and the transmitter cannot transmit a packet. The adapter is reset to reclaim packet buffers.
`Jumbo`	Number of packets transmitted that were larger than the standard Ethernet frame size (1,518 bytes).

LINK INFO section statistics

The following table describes the statistics in the LINK INFO section of the `ifstat` command output.

Statistic	Meaning
`Current state`	Current state of the interface: • `up` or `down`—The state of the link. • `cfg_down`—The interface is configured to the `down` status. • `enabling`—The interface is moving to the `up` status.
`Up to downs`	Number of times the link switched between the `up` status and the `down` status.

Statistic	Meaning
Auto	Operational state of autonegotiation: • on—Autonegotiation is enabled and succeeded. • off—Autonegotiation failed. This happens when the device to which the interface is connected has disabled autonegotiation or is incompatible with the interface. This might also indicate that the interface is in the down status.
Speed	Speed of link negotiated or set.
Duplex	Duplex of the link negotiated or set.
Flow control	The operational (autonegotiated) flow control setting.

Related tasks

Viewing or clearing network interface statistics on page 40

Statistics for Gigabit and 10 Gigabit Ethernet controllers T204, T210, and T320 interfaces

The ifstat command output displays several statistics when you use the command for the 10/100/1000 Ethernet controllers T204V and T204E, and the 10 Gigabit Ethernet controllers T210 and T320.

RECEIVE section statistics

The following table describes the statistics in the RECEIVE section of the ifstat command output when you use the command on the 10/100/1000 Ethernet controllers T204V and T204E, and the 10 Gigabit Ethernet controllers T210 and T320.

Statistic	Meaning
Frames/second	Rate of received frames per second.
Bytes/second	Rate of received bytes per second.
Errors/minute	Rate of errors (which led to frames being lost) per minute.
Discards/minute	Rate per minute of packets discarded due to unavailable resources.
Total frames	Total frames that are received on the interface.
Total bytes	Total bytes that are received on the interface.

Statistic	Meaning
`Total errors`	Total errors that occur on the interface.
`Total discards`	Total number of packets that were discarded even though no errors were detected. This number is a sum of the "No buffers", "Bus overruns", and "Queue overflows" statistics.
`Multi/broadcast`	Total number of multicast or broadcast packets received.
`Alignment errors`	Number of frames that are both misaligned and contain CRC errors.
`Non-primary u/c`	Number of Ethernet frames received for the partner's MAC address after a failover in an HA pair.
`Tag drop`	Number of tagged frames dropped on an interface that is not configured to support VLAN tagging.
`Vlan tag drop`	Number of tagged frames dropped that do not match the VLAN tags configured on the interface.
`Vlan untag drop`	Number of untagged frames dropped on an interface that is configured to be part of a VLAN.
`CRC errors`	Number of packets received with bad CRC.
`Runt frames`	Number of received frames that were less than the minimum size (64 bytes) and had a valid CRC.
`Long frames`	Number of received frames that were greater than the maximum size and had a valid CRC.
`Jabber`	Number of received frames that were greater than the maximum size and had a bad CRC.
`No buffer`	Number of times the driver could not allocate a buffer and a packet was dropped. This might happen when your storage system is very busy. If the count increases continually, it might indicate that a software component is not returning buffers.
`Jumbo`	Number of good packets received that were larger than the standard Ethernet packet size when jumbo frames are enabled.
`Pause Frames`	Number of MAC Control PAUSE frames sent to the link partner to inhibit transmission of data frames for a specified period of time. This can help the partner from overrunning the controller's receive buffers.

TRANSMIT section statistics

The following table describes the statistics in the TRANSMIT section of the `ifstat` command output when you use the command on the 10/100/1000 Ethernet controllers T204V and T204E, and the 10 Gigabit Ethernet controllers T210 and T320.

Statistic	Meaning
`Frames/second`	Rate of transmitted frames per second.
`Bytes/second`	Rate of transmitted bytes per second.
`Errors/minute`	Rate of errors (which led to frames being lost) per minute.
`Discards/minute`	Rate per minute of packets discarded due to unavailable resources.
`Total frames`	Total frames that are transmitted on the interface.
`Total bytes`	Total bytes that are transmitted on the interface.
`Total errors`	Total errors that occur on the interface.
`Total discards`	Total number of packets that were discarded even though no errors were detected. This number is a sum of the "No buffers" and "Queue overflows" statistics.
`Multi/broadcast`	Total number of multicast or broadcast packets transmitted.
`No buffers`	Number of times the driver failed to allocate a buffer for the transmit packet.
`Queue overflow`	Number of outgoing packets dropped because the driver's queue was full. It might indicate a system problem.
`Bus Underruns`	FIFO goes empty before an internal End-Of-Packet indicator is read.
`Pause Frames`	Number of MAC Control PAUSE frames sent to the link partner to inhibit transmission of data frames for a specified period of time. This can help the partner from overrunning the controller's receive buffers.

LINK INFO section statistics

The following table describes the statistics in the LINK INFO section of the `ifstat` command output when you use the command on the 10/100/1000 Ethernet controllers T204V and T204E, and the 10 Gigabit Ethernet controllers T210 and T320.

Statistic	Meaning
`Current state`	Current state of the interface: • `up` or `down`—The state of the link. • `cfg_down`—The interface is configured to the `down` status. • `enabling`—The interface is coming to the `up` status.
`Up to downs`	Number of times the link switched between the `up` status and the `down` status.
`Speed`	Speed of link negotiated or set.
`Duplex`	Duplex of the link negotiated or set.
`Flow control`	The operational (autonegotiated) flow control setting for T204 Gigabit Ethernet controller. The configured flow control setting for T210 and T320 10 Gigabit Ethernet controller. **Note:** Flow control cannot be autonegotiated on 10 Gigabit Ethernet controllers.

Statistics for the BGE 10/100/1000 Ethernet interface

The `ifstat` command output displays several statistics when you use the command on the BGE 10/100/1000 Ethernet interface.

RECEIVE section statistics

The following table describes the statistics in the RECEIVE section of the `ifstat` command output when you use the command on the BGE 10/100/1000 Ethernet interface.

Statistic	Meaning
`Frames/second`	Rate of received frames per second.
`Bytes/second`	Rate of received bytes per second.
`Errors/minute`	Rate of errors (which led to frames being lost) per minute.
`Discards/minute`	Rate per minute of packets discarded due to unavailable resources.
`Total frames`	Total frames that are received on the interface.
`Total bytes`	Total bytes that are received on the interface.

Statistic	Meaning
`Total errors`	Total errors that occur on the interface.
`Total discards`	Total number of packets that were discarded even though no errors were detected. This number is a sum of the "No buffers", "Bus overruns", and "Queue overflows" statistics.
`Multi/broadcast`	Total number of multicast or broadcast packets received.
`Alignment errors`	Number of frames that are both misaligned and contain CRC errors.
`Non-primary u/c`	Number of Ethernet frames received for the partner's MAC address after a failover in an HA pair.
`Tag drop`	Number of tagged frames dropped on an interface that is not configured to support VLAN tagging.
`Vlan tag drop`	Number of tagged frames dropped that do not match the VLAN tags configured on the interface.
`Vlan untag drop`	Number of untagged frames dropped on an interface that is configured to be part of a VLAN.
`CRC errors`	Number of packets received with bad CRC.
`Runt frames`	Number of received frames that were less than the minimum size (64 bytes) and had a valid CRC.
`Fragment`	Number of received frames that were less than the minimum size and had a bad CRC.
`Long frames`	Number of received frames that were greater than the maximum size and had a valid CRC.
`Jabber`	Number of received frames that were greater than the maximum size and had a bad CRC.
`No buffer`	Number of times the driver could not allocate a buffer and a packet was dropped. This might happen when your storage system is very busy. If the count increases continually, it might indicate that a software component is not returning buffers.
`Xon`	Number of XON frames received when receive or full flow control is enabled.
`Xoff`	Number of XOFF frames received when receive or full flow control is enabled.

Statistic	Meaning
`Jumbo`	Number of good packets received that were larger than the standard Ethernet packet size when jumbo frames are enabled.
`Ring full`	Not used. Ignore.
`Jumbo error`	Error detected while processing a jumbo packet. Packet is discarded.

TRANSMIT section statistics

The following table describes the statistics in the TRANSMIT section of the `ifstat` command output when you use the command on the BGE 10/100/1000 Ethernet interface.

Statistic	Meaning
`Frames/second`	Rate of transmitted frames per second.
`Bytes/second`	Rate of transmitted bytes per second.
`Errors/minute`	Rate of errors (which led to frames being lost) per minute.
`Discards/minute`	Rate per minute of packets discarded due to unavailable resources.
`Total frames`	Total frames that are transmitted on the interface.
`Total bytes`	Total bytes that are transmitted on the interface.
`Total errors`	Total errors that occur on the interface.
`Total discards`	Total number of packets that were discarded even though no errors were detected. This number is a sum of the "No buffers" and "Queue overflows" statistics.
`Multi/broadcast`	Total number of multicast or broadcast packets transmitted.
`No buffers`	Number of times the driver failed to allocate a buffer for the transmit packet.
`Queue overflow`	Number of outgoing packets dropped because the driver's queue was full. It might indicate a system problem.
`Max collisions`	Number of frames that were not transmitted because they encountered the maximum number of allowed collisions. Only valid in half-duplex mode.
`Single collision`	Number of frames that encountered exactly one collision. Only valid in half-duplex mode.

Statistic	Meaning
`Multi collisions`	Number of frames that encountered more than one collision, but less than the maximum allowed. Only valid in half-duplex mode.
`Late collisions`	Number of collisions that occurred outside the collision window. Only valid in half-duplex mode.
`Xon`	Number of XON frames transmitted when send or full flow control is enabled.
`Xoff`	Number of XOFF frames transmitted when send or full flow control is enabled.
`Jumbo`	Number of packets transmitted that were larger than the standard Ethernet packet size when jumbo frames are enabled.
`Deferred`	Number of frames for which the first transmission was delayed because the medium was busy.
`MAC Internal`	Number of frames not transmitted due to an internal MAC sublayer error.

LINK INFO section statistics

The following table describes the statistics in the LINK INFO section of the `ifstat` command output when you use the command on the BGE 10/100/1000 Ethernet interface.

Statistic	Meaning
`Current state`	Current state of the interface: • `up` or `down`—The state of the link. • `cfg_down`—The interface is configured to the `down` status. • `enabling`—The interface is coming to the `up` status.
`Up to downs`	Number of times the link switched between the `up` status and the `down` status.
`Speed`	Speed of link negotiated or set.
`Duplex`	Duplex of the link negotiated or set.
`Flow control`	The operational (autonegotiated) flow control setting.

Related tasks

Viewing or clearing network interface statistics on page 40

Statistics for 10 Gigabit Ethernet Controller IX1 - SFP+

The `ifstat` command output displays several statistics when you use the command for the Dual 10 Gigabit Ethernet controllers and 10 Gigabit Ethernet controllers.

RECEIVE section statistics

The following table describes the statistics in the RECEIVE section of the `ifstat` command output.

Statistic	Definition
`Frames/second`	Rate of received frames per second.
`Bytes/second`	Rate of received bytes per second.
`Errors/minute`	Rate of errors (which led to frames being lost) per minute.
`Discards/minute`	Rate per minute of packets discarded due to unavailable resources.
`Total frames`	Total frames that are received on the interface.
`Total bytes`	Total bytes that are received on the interface.
`Total errors`	Total errors that occur on the interface.
`Total discards`	Total number of packets that are discarded even though no errors were detected. This number is a sum of the "No buffers", "Bus overruns", and "Queue overflows" statistics.
`Multi/broadcast`	Total number of multicast or broadcast packets received.
`No buffers`	Number of times the driver could not allocate a buffer and a packet was dropped. This might occur when your storage system is very busy. If the count increases continually, it might indicate that a software component is not returning buffers.
`Non-primary u/c`	Number of Ethernet frames received for the partner's MAC address after a failover in an HA pair.
`Tag drop`	Number of tagged frames dropped on an interface that is not configured to support VLAN tagging.
`Vlan tag drop`	Number of tagged frames dropped that do not match the VLAN tags configured on the interface.
`Vlan untag drop`	Number of untagged frames dropped on an interface that is configured to be part of a VLAN.
`CRC errors`	Number of packets received with bad CRC.

Statistic	Definition
`Runt frames`	Number of received frames that were less than the minimum size (64 bytes) and had a valid CRC.
`Fragment`	Number of received frames that were less than the minimum size and had a bad CRC.
`Long frames`	Number of received frames that were greater than the maximum size and had a valid CRC.
`Jabber`	Number of received frames that were greater than the maximum size and had a bad CRC.
`Bus overruns`	Number of times the adapter's receive FIFO overflowed and a packet was dropped. This occurs when the bus is very busy and the adapter cannot transfer data into host memory. This might also occur when your storage system CPU is very busy and cannot process the received packets quickly.
`Xon`	Number of XON frames received when receive or full flow control is enabled.
`Xoff`	Number of XOFF frames received when receive or full flow control is enabled.
`Jumbo`	Number of good packets received that are larger than the standard Ethernet packet size when jumbo frames are enabled.

TRANSMIT section statistics

The following table describes the statistics in the TRANSMIT section of the `ifstat` command output.

Statistic	Meaning
`Frames/second`	Rate of transmitted frames per second.
`Bytes/second`	Rate of transmitted bytes per second.
`Errors/minute`	Rate of errors (which led to frames being lost) per minute.
`Discards/minute`	Rate per minute of packets discarded due to unavailable resources.
`Pktlen`	Number of transmit packets passed to the driver with an incorrect length. Should be 0.
`Total frames`	Total frames that are transmitted on the interface.
`Total bytes`	Total bytes that are transmitted on the interface.

Statistic	Meaning
`Total errors`	Total errors that occur on the interface.
`Total discards`	Total number of discarded packets even though no errors are detected. This number is a sum of the "No buffers" and "Queue overflows" statistics.
`Multi/broadcast`	Total number of multicast or broadcast packets transmitted.
`No buffers`	Number of times the driver failed to allocate a buffer for the transmit packet.
`Queue overflows`	Number of outgoing packets dropped because the driver's queue was full. It might indicate a system problem.
`Xon`	Number of XON frames transmitted when send or full flow control is enabled.
`Xoff`	Number of XOFF frames transmitted when send or full flow control is enabled.
`Timeout`	Number of times the adapter's transmitter was hung and the adapter had to be reset. This might occur when the cable is pulled and the transmitter cannot transmit a packet. The adapter is reset to reclaim packet buffers.
`Jumbo`	Number of packets transmitted that are larger than the standard Ethernet frame size (1,518 bytes).

LINK INFO section statistics

The following table describes the statistics in the LINK INFO section of the `ifstat` command output.

Statistic	Meaning
`Current state`	Current state of the interface: • `up` or `down`—The state of the link. • `cfg_down`—The interface is configured to the `down` status. • `enabling`—The interface is moving to the `up` status.
`Up to downs`	Number of times the link is switched between the `up` status and the `down` status.
`Speed`	Speed of link negotiated or set.
`Duplex`	Duplex of the link negotiated or set.

Statistic	Meaning
`Flow control`	The configured flow control setting, as flow control is not autonegotiated on this interface.

Statistics for Dual 10G Ethernet Controller CNA - SFP+

The `ifstat` command output displays several statistics when you use the command for the Dual 10G Ethernet controllers.

RECEIVE section statistics

The following table describes the statistics in the RECEIVE section of the `ifstat` command output.

Statistic	Definition
`Frames/second`	Rate of received frames per second.
`Bytes/second`	Rate of received bytes per second.
`Errors/minute`	Rate of errors (which led to frames being lost) per minute.
`Discards/minute`	Rate of packets discarded per minute due to unavailable resources.
`Total frames`	Total frames received by the interface.
`Total bytes`	Total bytes received by the interface.
`Total errors`	Total errors that occur on the interface.
`Total discards`	Total number of packets that were discarded inspite of undetected errors. This number is a sum of the "No buffers", "Bus overruns", and "Queue overflows" statistics.
`Multi/broadcast`	Total number of multicast or broadcast packets received.
`No buffers`	Number of times the driver could not allocate a buffer and a packet was dropped. This might occur when your storage system is busy. If the count increases continually, it might indicate that a software component is not returning buffers.
`Non-primary u/c`	Number of Ethernet frames received for the partner's MAC address after a failover in an HA pair.
`Tag drop`	Number of tagged frames dropped on an interface that is not configured to support VLAN tagging.
`Vlan tag drop`	Number of tagged frames dropped that do not match the VLAN tags configured on the interface.

Statistic	Definition
`Vlan untag drop`	Number of untagged frames dropped on an interface that is configured to be part of a VLAN.
`CRC errors`	Number of packets received with bad CRC.
`Alignment errors`	Number of frames that are both misaligned and contain CRC errors.
`Runt frames`	Number of received frames that were less than the minimum size (64 bytes) and had a valid CRC.
`Long frames`	Number of received frames that were greater than the maximum size and had a valid CRC.
`Code error`	Number of received packets/ frames with code errors.
`Preamble error`	Number of received packets/ frames with preamble errors.
`Frame length error`	Number of received frames with frame length errors.
`Received VLAN frames`	Number of received VLAN frames.

TRANSMIT section statistics

The following table describes the statistics in the TRANSMIT section of the `ifstat` command output.

Statistic	Meaning
`Frames/second`	Rate of transmitted frames per second.
`Bytes/second`	Rate of transmitted bytes per second.
`Errors/minute`	Rate of errors (which led to frames being lost) per minute.
`Discards/minute`	Rate of packets discarded per minute due to unavailable resources.
`Total frames`	Total frames that were transmitted on the interface.
`Total bytes`	Total bytes that were transmitted on the interface.
`Total errors`	Total errors that occur on the interface.
`Total discards`	Total number of discarded packets even though no errors were detected. This number is a sum of the "No buffers" and "Queue overflows" statistics.
`Multi/broadcast`	Total number of multicast or broadcast packets transmitted.

Statistic	Meaning
`No buffers`	Number of times the driver failed to allocate a buffer for the transmit packet.
`Queue overflows`	Number of outgoing packets dropped because the driver's queue was full. It might indicate a system problem.
`Backplane error`	Number of PCI backplane errors detected.
`Padded sent`	Number of 16-60 byte frames, padded and sent.
`Too long frames`	Number of frames that were too long but were sent anyway.
`Too short frames`	Number of frames that were too short and illegal and were not sent.
`Unmatched length`	Total descriptor length that does not match transfer length.

LINK INFO section statistics

The following table describes the statistics in the LINK INFO section of the `ifstat` command output.

Statistic	Meaning
`Current state`	Current state of the interface: • `up` or `down`—The state of the link. • `cfg_down`—The interface is configured to the `down` status. • `enabling`—The interface is moving to the `up` status.
`Up to downs`	Number of times the link is switched between the `up` status and the `down` status.
`Speed`	Speed of link negotiated or set.
`Duplex`	Duplex of the link negotiated or set.
`Flow control`	The configured flow control setting, as flow control is not autonegotiated on this interface.

Statistics for Quad Gigabit Ethernet Controller 82850

The `ifstat` command output displays several statistics when you use the command for the Quad Gigabit Ethernet Controllers.

RECEIVE section statistics

The following table describes the statistics in the RECEIVE section of the `ifstat` command output.

Statistic	Definition
`Frames/second`	Rate of received frames per second.
`Bytes/second`	Rate of received bytes per second.
`Errors/minute`	Rate of errors (which led to frames being lost) per minute.
`Discards/minute`	Rate of packets discarded per minute due to unavailable resources.
`Total frames`	Total frames received by the interface.
`Total bytes`	Total bytes received by the interface.
`Total errors`	Total errors that occur on the interface.
`Total discards`	Total number of packets that were discarded inspite of undetected errors. This number is a sum of the "No buffers", "Bus overruns", and "Queue overflows" statistics.
`Multi/broadcast`	Total number of multicast or broadcast packets received.
`No buffers`	Number of times the driver could not allocate a buffer and a packet was dropped. This might occur when your storage system is busy. If the count increases continually, it is an indication that a software component is not returning buffers.
`Non-primary u/c`	Number of Ethernet frames received for the partner's MAC address after a failover in an HA pair.
`Tag drop`	Number of tagged frames dropped on an interface that is not configured to support VLAN tagging.
`Vlan tag drop`	Number of tagged frames dropped that do not match the VLAN tags configured on the interface.
`Vlan untag drop`	Number of untagged frames dropped on an interface that is configured to be part of a VLAN.

Statistic	Definition
`CRC errors`	Number of packets received with bad CRC.
`Runt frames`	Number of received frames that were less than the minimum size (64 bytes) and had a valid CRC.
`Fragment`	Number of received frames that were less than the minimum size and had a bad CRC.
`Long frames`	Number of received frames that were greater than the maximum size and had a valid CRC.
`Jabber`	Number of received frames that were greater than the maximum size and had a bad CRC.
`Alignment errors`	Number of frames that are misaligned and contain CRC errors.
`Bus overruns`	Number of times the adapter's receive FIFO overflowed and a packet was dropped. This occurs when the bus is too busy and the adapter cannot transfer data into host's memory. This might also occur when your storage system CPU is too busy and cannot process the received packets fast enough.
`Xon`	Number of XON frames received when receive or full flow control is enabled.
`Xoff`	Number of XOFF frames received when receive or full flow control is enabled.
`Jumbo`	Number of good packets received that were larger than the standard Ethernet packet size when jumbo frames are enabled.

TRANSMIT section statistics

The following table describes the statistics in the TRANSMIT section of the `ifstat` command output.

Statistic	Meaning
`Frames/second`	Rate of transmitted frames per second.
`Bytes/second`	Rate of transmitted bytes per second.
`Errors/minute`	Rate of errors (which led to frames being lost) per minute.
`Discards/minute`	Rate of packets discarded per minute due to unavailable resources.
`Total frames`	Total frames that were transmitted on the interface.
`Total bytes`	Total bytes that were transmitted on the interface.

Statistic	Meaning
`Total errors`	Total errors that occur on the interface.
`Total discards`	Total number of discarded packets even though no errors were detected. This number is a sum of the "No buffers" and "Queue overflows" statistics.
`Multi/broadcast`	Total number of multicast or broadcast packets transmitted.
`Queue overflows`	Number of outgoing packets dropped because the driver's queue was full and might indicate a system problem.
`No buffers`	Number of times the driver failed to allocate a buffer for the transmit packet.
`Max collisions`	Number of frames that encountered more than one collision, but less than the maximum allowed. Only valid in half-duplex mode.
`Single collision`	Number of frames that encountered exactly one collision. Only valid in half-duplex mode.
`Multi collisions`	Number of frames that encountered more than one collision, but less than the maximum allowed. Only valid in half-duplex mode.
`Late collisions`	Number of collisions that occurred outside the collision window. Only valid in half-duplex mode.
`Xon`	Number of XON frames transmitted when send or full flow control is enabled.
`Xoff`	Number of XOFF frames transmitted when send or full flow control is enabled.
`Jumbo`	Number of packets transmitted that were larger than the standard Ethernet frame size (1,518 bytes).

LINK INFO section statistics

The following table describes the statistics in the LINK INFO section of the `ifstat` command output.

Statistic	Meaning
`Current state`	Current state of the interface: • `up` or `down`—The state of the link. • `cfg_down`—The interface is configured to the `down` status. • `enabling`—The interface is moving to the `up` status.
`Up to downs`	Number of times the link is switched between the `up` status and the `down` status.
`Speed`	Speed of link negotiated or set.
`Duplex`	Duplex of the link negotiated or set.
`Flow control`	Operational (autonegotiated) flow control setting.

Ways to improve your storage system's performance

You can improve your storage system's performance by performing certain configuration procedures, such as using interface groups, correcting duplex mismatches, and upgrading to Ethernet interfaces.

The following configuration procedures might improve the performance of your storage system:

- Using static or dynamic multimode interface groups to aggregate the bandwidth of multiple interfaces
- Using jumbo frames with your network interfaces to reduce CPU processing overhead
- Upgrading to a faster network interface

  You can increase the storage system's performance by upgrading to a faster network interface (10 Gigabit Ethernet interfaces).
- Correcting duplex mismatches on 10Base-T or 100Base-T Ethernet networks

  On 10Base-T or 100Base-T Ethernet networks, the speed and duplex settings for the interfaces at both ends of a link must match exactly. You can use the `ifconfig` *`interface`* command to check the duplex setting of your storage system’s interface.

  If the setting is to autonegotiate, the `ifconfig` command displays a setting that begins with `auto` (for example, `auto-100tx-fd-up`). Otherwise, the `ifconfig` command displays the configured media type setting (for example, `100tx-fd-up`).

  Note: If one end of the link is set to autonegotiate, the other end must also be set to autonegotiate; otherwise, a mismatch might occur. You can determine the negotiated setting with the `ifstat` command.
- Using iSCSI multiconnection sessions to balance the load across interfaces

  For each iSCSI session, multiple connections are created. The number of allowed connections is negotiated during login and session creation. While it is possible to create multiple connections over a single physical interface, it is best to use multiple physical interfaces for bandwidth enhancement.
- Enabling fast path on your storage system

  Fast path provides load balancing by sending responses on the same network interface that receives the incoming requests and improved performance by skipping routing table lookups.
- Blocking data traffic on the dedicated management interface

  In storage systems with dedicated management interface, e0M, you should block certain types of data traffic on e0M, such as SnapMirror transfers and SnapVault transfers, and other data transfers that use the file access protocols such as CIFS, NFS, and iSCSI. If the low-bandwidth management interface is configured to serve data traffic, system performance might be affected. Therefore, e0M should be used only for Data ONTAP management activities such as running a Telnet, RSH, or SSH session.

Related concepts

Static multimode interface group on page 97
Dynamic multimode interface group on page 98
What jumbo frames are on page 21

Related tasks

Specifying a media type for a network interface on page 27
Enabling or disabling fast path on page 58

IP port usage on a storage system

The Data ONTAP services file is available in the `/etc` directory. The `/etc/services` file is in the same format as its corresponding UNIX system's `/etc/services` file.

Host identification

Although some port scanners are able to identify storage systems as storage systems, other port scanners report storage systems as UNIX systems if they support NFS or as Windows systems if they support CIFS. There are several services that are not currently listed in the `/etc/services` file.

The following table gives a sample content of the `/etc/services` file.

Service	Port/ Protocol	Description
`ftp-data`	`20/tcp`	`# File transfer protocol`
`ftp`	`21/tcp`	`# File transfer protocol`
`ssh`	`22/tcp`	`# SecureAdmin rsh replacement`
`telnet`	`23/tcp`	`# Remote login (insecure)`
`smtp`	`25/tcp`	`# outbound connections for autosupport`
`time`	`37/tcp`	`# Time Service`
`time`	`37/udp`	`# Time Service`
`domain`	`53/udp`	`# DNS - outbound only`
`domain`	`53/tcp`	`# DNS zone transfers - unused`
`dhcps`	`67/udp`	`# DHCP server - outbound only`
`dhcp`	`68/udp`	`# DHCP client - only first-time setup`
`tftp`	`69/udp`	`# Trivial FTP - for netboot support`

Service	Port/ Protocol	Description
kerberos	88/udp	# Kerberos 5 - outbound only
kerberos	88/tcp	# Kerberos 5 - outbound only
portmap	111/udp	# aka rpcbind, used for NFS
portmap	111/tcp	# aka rpcbind, used for NFS
nntp	119/tcp	# unused, shouldn't be listed here.
ntp	123/tcp	# Network Time Protocol
ntp	123/udp	# Network Time Protocol
netbios-name	137/udp	# NetBIOS nameserver - for CIFS
netbios-dg	138/udp	# NetBIOS datagram service - for CIFS
netbios-ssn	139/tcp	# NetBIOS service session - for CIFS
cifs-tcp	445/tcp	# CIFS over TCP with NetBIOS framing
snmp	161/udp	# For Data Fabric Manager or other such tools
ldap	389/tcp	# LDAP session
https	443/tcp	# SecureAdmin/SSL
shell	514/tcp	# rsh, insecure remote command execution.
syslog	514/udp	# outbound only
route	520/udp	# for RIP routing protocol
ldap-ssl	636/tcp	# LDAP over SSL
kerberos-sec	750/udp	# outbound only, if at all

Service	Port/ Protocol	Description
kerberos-sec	750/tcp	# outbound only, if at all
nfsd	2049/udp	# primary NFS service
nfsd	2049/tcp	# primary NFS service
nrv	2050/tcp	# Remote Volume protocol, used in FlexCache and Restore-On-Demand
iscsi-target	3260/tcp	# iSCSI target port
nlockmgr	4045/tcp	# NLM
nlockmgr	4045/udp	# NLM
mountd	4046/tcp	# NFS mountd protocol
mountd	4046/udp	# NFS mountd protocol
status	4047/tcp	# CTP state
status	4047/udp	# CTP state
pcnfsd	4048/tcp	# PCNFS protocol
pcnfsd	4048/udp	# PCNFS protocol
rquotad	4049/udp	# NFS rquotad protocol
ndmp	10000/tcp	# for network backups
sm-ics	10565/tcp	# Snapmirror Multipath
snapmirror	10566/tcp	# SnapMirror and SnapVault
sm-sync-block	10567/tcp	# Snapmirror Sync Block Data
sm-sync-trans	10568/tcp	# Snapmirror Sync Transaction Data
sm-sync-ctrl	10569/tcp	# Snapmirror Sync Control Data

/etc/services NNTP and TTCP ports

The NNTP and TTCP ports are not used by your storage system and should never be detected by a port scanner.

NFS-enabled ports

Some ports (port numbers in the 600 range) on the storage system are NFS-enabled.

The following ports are found on the storage system with NFS-enabled.

UDP	602	NFS mount daemon (`mountd`)
TCP	603	NFS mount daemon (`mountd`)
UDP	604	NFS status daemon (`statd`, `statmon`)
TCP	605	NFS status daemon (`statd`, `statmon`)
UDP	606	NFS lock manager (`lockd`, `nlockmgr`)
TCP	607	NFS lock manager (`lockd`, `nlockmgr`)
UDP	608	NFS quota daemon (`quotad`, `rquotad`)

On other systems, the ports appear as follows:

UDP	611	NFS mount daemon (`mountd`)
TCP	612	NFS mount daemon (`mountd`)
UDP	613	NFS status daemon (`statd`, `statmon`)
TCP	614	NFS status daemon (`statd`, `statmon`)
UDP	615	NFS lock manager (`lockd`, `nlockmgr`)
TCP	616	NFS lock manager (`lockd`, `nlockmgr`)
UDP	617	NFS quota daemon (`quotad`, `rquotad`)

The following command on UNIX systems obtains the correct information by querying the port mapper on port 111:

```
toaster# rpcinfo -p sys1

   program vers proto   port  service
    100011    1   udp    608  rquotad
    100021    4   tcp    607  nlockmgr
    100021    3   tcp    607  nlockmgr
```

```
   100021    1   tcp    607  nlockmgr
   100021    4   udp    606  nlockmgr
   100021    3   udp    606  nlockmgr
   100021    1   udp    606  nlockmgr
   100024    1   tcp    605  status
   100024    1   udp    604  status
   100005    3   tcp    603  mountd
   100005    2   tcp    603  mountd
   100005    1   tcp    603  mountd
   100005    3   udp    602  mountd
   100005    2   udp    602  mountd
   100005    1   udp    602  mountd
   100003    3   udp   2049  nfs
   100003    2   udp   2049  nfs
   100000    2   tcp    111  rpcbind
   100000    2   udp    111  rpcbind
```

Note: The port numbers listed for `mountd`, `statd`, `lockd`, and `quotad` are not committed port numbers. These services can be running on other ports of the storage systems. Because the system selects these port numbers at random when it boots, they are not listed in the `/etc/services` file.

Ports not listed in /etc/services

Some ports appear in a port scan but are not listed in the `/etc/services` file, for example, TCP ports 22 and 443.

The following ports appear in a port scan but are not listed in the `/etc/services` file.

Protocol	Port	Service
TCP	22	SSH (SecureAdmin)
TCP	443	SSL (SecureAdmin)
TCP	3260	iSCSI-Target

Note: Disable the open ports that you do not need.

FTP

File Transfer Protocol (FTP) uses TCP ports 20 and 21.

If you use FTP to transfer files to and from your storage system, the FTP port is required; otherwise, use the following CLI command to disable the FTP port:

```
options ftpd.enable off
```

FTP is not a secure protocol for two reasons:

- When users log in to the system, user names and passwords are transmitted over the network in clear text format that can easily be read by a packet sniffer program.
  These user names and passwords can then be used to access data and other network resources. You should establish and enforce policies that prevent the use of the same passwords to access storage systems and other network resources.
- FTP server software used on platforms other than storage systems contains serious security-related flaws that allow unauthorized users to gain administrative (root) access and control over the host.

Starting with Data ONTAP 7.3.1, FTP over IPv6 is supported.

For a detailed description of the FTP support for your storage system, see the *Data ONTAP File Access and Protocols Management Guide for 7-Mode*.

SSH

Secure Shell (SSH) protocol is a secure replacement for RSH and runs on TCP port 22. This port appears in a port scan only if the SecureAdmin software is installed on your storage system.

There are three commonly deployed versions of the SSH protocol:

- SSH version 1—is secure than RSH or Telnet, but is vulnerable to TCP session attacks.
  This vulnerability to attack lies in the SSH protocol version 1 itself and not in the associated storage system products.
- SSH version 2—has a number of feature improvements over SSH version 1 and is less vulnerable to attacks.
- SSH version 1.5—is used to identify clients or servers that support both SSH versions 1 and 2.

To disable SSH support or to close TCP port 22, you must use the following CLI command:

```
secureadmin disable ssh
```

Telnet

Telnet is used for administrative control of your storage system and uses TCP connections on port 23. Telnet is more secure than RSH, as secure as FTP, and less secure than SSH or Secure Socket Layer (SSL).

Telnet is less secure than SSH and SSL because:

- When users log in to a system, such as your storage system, user names and passwords are transmitted over the network in clear text format.
  Clear text format can be read by an attacker by using a packet sniffer program. The attacker can use these user names and passwords to log in to your storage system and execute unauthorized administrative functions, including destruction of data on the system. If administrators use the

same passwords on your storage system as they do on other network devices, the attacker can use these passwords to access the resources of the storage system as well.

Note: To reduce the potential for attack, you must establish and enforce policies preventing administrators from using the same passwords on your storage system that they use to access other network resources.

- Telnet server software used on other platforms (typically in UNIX environments) have serious security-related flaws that allow unauthorized users to gain administrative (root) control over the host.

Telnet is also vulnerable to the same type of TCP session attacks as SSH protocol version 1. However, TCP session attacks are less common because a packet sniffing attack is easier.

To disable Telnet, you must set `options telnet.enable` to `off`.

Starting with Data ONTAP 7.3.1, Telnet supports IPv6. However, if you have enabled the IPv6 option when the storage system is in operation (not during setup), you must restart the Telnet service. That is, you need to turn off and then turn on the Telnet service for connecting over IPv6.

SMTP

Simple Mail Transport Protocol (SMTP) uses TCP port 25. Your storage system does not listen on this port but makes outgoing connections to mail servers using this protocol when sending AutoSupport e-mail.

Time service

Your storage system supports the Network Time Protocol (NTP) for time synchronization.

To synchronize the system time with a network time server, you must ensure that the `timed.enable` option is set to `on`, the `timed.proto` option is set to `ntp`, and the `timed.servers` option is set with at least one valid time server.

Note: Both controllers in an HA pair should have time synchronization enabled to ensure time consistency in the event of controller failover.

For more information about synchronizing the system time, see the *Data ONTAP System Administration Guide for 7-Mode*.

DNS

The Domain Name System (DNS) uses UDP port 53 and TCP port 53. Your storage system does not typically listen on these ports because it does not run a domain name server. However, if DNS is

enabled on your storage system, it makes outgoing connections using UDP port 53 for host name and IP address lookups.

The storage system never uses TCP port 53 because this port is used explicitly for communication between DNS servers. Outgoing DNS queries by your storage system are disabled by turning off DNS support. Turning off DNS support protects against receiving bad information from another DNS server.

Because your storage system does not run a domain name server, the name service must be provided by one of the following:

- Network information service (NIS)
- An `/etc/hosts` file
- Replacement of host names in the configuration files (such as `/etc/exports`, `/etc/usermap.cfg`, and so on) with IP addresses

DNS must be enabled for participation in an Active Directory domain.

DHCP

Clients broadcast messages to the entire network on UDP port 67 and receive responses from the Dynamic Host Configuration Protocol (DHCP) server on UDP port 68. The same ports are used for the BOOTP protocol.

DHCP is used only for the first-time setup of your storage system. Detection of DHCP activity on your storage system by a port scan other than the activity during the first-time setup indicates a serious configuration or software error.

TFTP

Trivial File Transfer Protocol (TFTP) uses TCP port 69. It is used mostly for booting UNIX or UNIX-like systems that do not have a local disk (this process is also known as netbooting) and for storing and retrieving configuration files for devices such as Cisco routers and switches.

Transfers are not secure on TFTP because it does not require authentication for clients to connect and transfer files.

Your storage system's TFTP server is not enabled by default. When TFTP is enabled, the administrator must specify a directory to be used by TFTP clients, and these clients cannot access other directories. Even within the TFTP directory, access is read-only. TFTP should be enabled only if necessary. You can disable TFTP using the following option:

```
options tftpd.enable off
```

You can configure the maximum number of simultaneous connections handled by the TFTP server by using the `tftpd.max_connections` option. The default number of TFTP connections is 8. The maximum number of connections supported is 32.

HTTP

Hypertext Transport Protocol (HTTP) runs on TCP port 80 and is the protocol used by Web browsers to access Web pages.

Your storage system uses HTTP to access files.

Starting with Data ONTAP 7.3.1, HTTP over IPv6 is supported and can be used for file access.

HTTP is not vulnerable to security attacks because it provides read-only access to documents by unauthenticated clients. Although authentication is not typically used for file access, it is frequently used for access to restricted documents or for administration purposes. The authentication methods defined by HTTP send credentials, such as user names and passwords, over the network without encryption. The SecureAdmin product is provided with SSL support to overcome this shortcoming.

Note: In versions earlier than Data ONTAP 7.0, your storage system listens for new connections (by default, set to TCP port 80) even when HTTP is not licensed.. However, starting with Data ONTAP 7.0, you can stop your storage system from listening for new connections by setting the options `httpd.enable` and `httpd.admin.enable` to `off`. If either of the options is set to `on`, your storage system will continue to listen for new connections.

Kerberos

There are four Kerberos ports in the `/etc/services` file: TCP port 88, UDP port 88, TCP port 750, and UDP port 750. These ports are used only for outbound connections from your storage system. Your storage system does not run Kerberos servers or services and does not listen on these ports.

Kerberos is used by your storage system to communicate with the Microsoft Active Directory servers for both CIFS authentication and, if configured, NFS authentication.

NFS

Network File System (NFS) is used by UNIX clients for file access. NFS uses port 2049.

NFSv3 and NFSv2 use the portmapper service on TCP or UDP port 111. The portmapper service is consulted to get the port numbers for services used with NFSv3 or NFSv2 protocols such as mountd, statd, and nlm. NFSv4 does not require the portmapper service.

NFSv4 provides the delegation feature that enables your storage system to grant local file access to clients. To delegate, your storage system sets up a separate connection to the client and sends callbacks on it. To communicate with the client, your storage system uses one of the reserved ports (port numbers less than 1024). To initiate the connection, the client registers the callback program on a random port and informs the server about it.

With delegations enabled, NFSv4 is not firewall-friendly because several other ports need to be opened up as well.

Starting with Data ONTAP 7.3.1, IPv6 over NFS is supported.

You can disable the TCP and UDP ports by setting the `nfs.tcp.enable` and `nfs.udp.enable` options to `off`.

To disable NFS, you should use the `nfs off` command.

CIFS

Common Internet File Service (CIFS) is the successor to the server message block (SMB) protocol. CIFS is the primary protocol used by Windows systems for file sharing.

CIFS uses UDP ports 137 and 138, and TCP ports 139 and 445. Your storage system sends and receives data on these ports while providing CIFS service. If it is a member of an Active Directory domain, your storage system must also make outbound connections destined for DNS and Kerberos.

Starting with Data ONTAP 7.3.1, CIFS over IPv6 is supported. CIFS over IPv6 uses only port 445. Ports 137, 138, and 139 are used by NetBIOS, which does not support IPv6.

CIFS is required for Windows file service. You can disable CIFS by issuing the `cifs terminate` command on your storage system console.

Note: If you disable CIFS, be aware that your storage system's `/etc/rc` file can be set up to automatically enable CIFS again after a reboot.

SSL

The Secure Sockets Layer (SSL) protocol provides encryption and authentication of TCP connections. Data ONTAP supports SSLv2, SSLv3, and Transport Layer Security (TLS) version 1.0. You should use TLSv1.0 or SSLv3 because it offers better security than previous SSL versions.

When SecureAdmin is installed and configured on your storage system, it listens for SSL connections on TCP port 443.

You can enable or disable SSL with the following command:

```
secureadmin {enable|disable} ssl
```

For TLS to be used for communication, both the client requesting the connection and the storage system must support TLS.

TLS is disabled by default, and setting up SSL does not automatically enable TLS. Before enabling TLS, ensure that SSL has been set up and enabled. To enable or disable TLS, enter the following command:

```
options tls.enable {on|off}
```

SNMP

Simple Network Management Protocol (SNMP) is an industry-standard protocol used for remote monitoring and management of network devices over UDP port 161.

SNMP is not secure because of the following reasons:

- Instead of using encryption keys or a user name and password pair, SNMP uses a community string for authentication. The community string is transmitted in clear text format over the network, making it easy to capture with a packet sniffer.

  Within the industry, devices are typically configured at the factory to use `public` as the default community string. The public password allows users to make queries and read values but does not allow users to invoke commands or change values. Some devices are configured at the factory to use `private` as the default community string, allowing users full read-write access.
- Even if you change the read and write community string on a device to something other than `private`, an attacker can easily learn the new string by using the read-only `public` community string and asking the router for the read-write string.

There are three versions of SNMP:

- SNMPv1 is the original protocol and is not commonly used.
- SNMPv2 is identical to SNMPv1 from a network protocol standpoint and is vulnerable to the same security problems. The only differences between the two versions are in the messages sent, messages received, and types of information. These differences are not important from a security perspective.
- SNMPv3 is the latest protocol version and includes security improvements but is difficult to implement and many vendors do not yet support it. SNMPv3 supports several different types of network encryption and authentication schemes. It allows for multiple users, each with different permissions, and solves SNMPv1 security problems while maintaining an important level of compatibility with SNMPv2.

SNMP is required if you want to monitor a storage system through an SNMP monitoring tool, such as DataFabric Manager. The SNMP implementation in the storage system allows read-only access. Regardless of the community string used, the user cannot issue commands or change variables using SNMP on your storage system.

You should use the `snmp.access` option to restrict SNMP access to a named set of trusted hosts.

You can disable SNMP entirely by setting the `snmp.enable` option to `off` to disable SNMP entirely.

The `snmp community delete` and `snmp community add` commands are used to change the community string to something other than the default value.

RSH

Remote Shell (RSH) protocol is used for remote command execution. It is less secure than TFTP and uses TCP port 514.

RSH is not secure because passwords are not required for login and commands are easy to misconfigure. Therefore, you should disable RSH by setting the `rsh.enable` option to `off`.

You should use the SSH supplied with SecureAdmin for remote command execution and login. If this is not possible, Telnet is preferred to RSH.

If RSH is the only alternative, follow these guidelines when using RSH:

- Specify only secure, trusted hosts in the `/etc/hosts.equiv` file.
- Always use IP addresses rather than host names in the `/etc/hosts.equiv` file.
- Always specify a single IP address with a single user name on each line in `/etc/hosts.equiv` file.
- Use the `rsh.access` option instead of the `trusted.hosts` option for access control.
- Make sure the `ip.match_any_ifaddr` option is set to `off`.

Syslog

Your storage system sends messages to hosts specified by the user in the `/etc/syslog.conf` file by using the syslog protocol on UDP port 514. It does not listen on this port, nor does it act as a syslog server.

The routed daemon

The routed daemon, `routed`, listens on UDP port 520. It receives broadcast messages, using the Routing Information Protocol (RIP), from routers or other hosts. These messages are used by your storage system to update its internal routing tables to determine the optimal network interfaces for each destination.

Your storage system does not broadcast RIP messages containing routes because Data ONTAP cannot act as a router.

RIP is not a secure protocol because an attacker can easily send artificial RIP messages and cause hosts running the routed daemon (such as your storage system) to redirect network traffic to the attacker. The attacker can then receive and shift this traffic for passwords and other information and send it on to the actual destination, where the intrusion is undetected. This method can also be used as a starting point for TCP session attacks.

Because of these security issues, you must use static routes (those set up using the `route` command on your storage system) instead of using the routed daemon.

If you require dynamic routing in untrusted environments, you must use RIPv2 with authentication. This is because the `routed` daemon poses a security risk if you use RIPv1 in untrusted environments.

NDMP

Network Data Management Protocol (NDMP) runs on TCP port 10000 and is used primarily for backup of network-attached storage (NAS) devices, such as storage systems.

The protocol defines three authentication methods:

- NONE—allows authentication without restriction
- TEXT—sends a clear text password over the network, similar to Telnet or FTP
- MD5—uses the MD5 message digest algorithm along with a challenge-response message exchange to implement a secure login mechanism

Your storage system supports both the TEXT and MD5 authentication methods. Most NDMP-enabled backup software uses MD5 by default.

To entirely disable the TEXT authentication method, you should set the `ndmpd.authtype` option to `challenge`.

To restrict NDMP commands to certain authorized backup hosts, you should use the `ndmp.access` option.

Regardless of the authentication method used, NDMP sends backup data in decrypted format over the network, as does most other backup software. A separate network optimized for backup is a common means to increase performance while retaining data security.

To disable NDMP, you should set the `ndmp.enable` option to `off`.

SnapMirror and SnapVault

SnapMirror and SnapVault use TCP port 10566 for data transfer. Network connections are always initiated by the destination system; that is, SnapMirror and SnapVault *pull* data rather than *push* data.

Authentication is minimal with both SnapMirror and SnapVault. To restrict inbound TCP connections on port 10566 to a list of authorized hosts or IP addresses, you should configure the `snapmirror.access` or `snapvault.access` option. When a connection is established, the destination storage system communicates its host name to the source storage system, which then uses this host name to determine if a transfer is allowed. You should confirm a match between the host name and its IP address. To confirm that the host name and the IP address match, you should set the `snapmirror.checkip.enable` option to `on`.

To disable SnapMirror, you should set the `snapmirror.enable` option to `off`. To disable SnapVault, you should set the `snapvault.enable` option to `off`.

Error codes for the netdiag command

Network error codes are generated by the `netdiag` command. They describe the network problems and suggest the actions that you can take.

The following table lists some network error codes, describes the problems that the error codes point to, and suggests actions that you can take to fix the problems.

Note: Only a small fraction of the possible network error messages are presented in the following table. If you receive any problem code not listed in this table, contact your technical support.

Error code	Description	Recommended actions
201	Link not detected.	Complete the following steps until you detect a link: **1.** Ensure that the cable is connected between the switch port and your storage system interface, and that both ends are securely attached. **2.** Ensure that the switch port and interface are both configured to the `up` status, and one of the following is true: • Autonegotiation is enabled on both sides • Autonegotiation is disabled on both sides, and the duplex and speed settings match **3.** Because the switch port, cable, or NIC might be faulty, replace them, one by one, to locate the fault. **4.** If the problem persists, contact your technical support.
203	No link is detected because of a speed mismatch.	Change the interface configuration or peer switch port configuration to match the speed.
204	The interface is not configured to the `up` status.	Configure the interface state to the `up` status.
205	Duplex mismatch.	Change the interface or peer switch port duplex setting so that they match.
206	Link capacity problem.	Upgrade to a faster interface.

Error code	Description	Recommended actions
207	The interface is not transmitting or receiving.	Complete the following steps: **1.** Pull the network cable out from the network interface card. **2.** Reinsert the cable. **3.** Use `ifstat` to display statistics. • Link errors, such as CRC, are caused by a faulty switch port, cable, or NIC; replace them one by one to locate the fault. • Out-of-resource errors are caused by heavy loads. **4.** If the problem persists, contact your technical support.
208	Excessive I/O errors.	Complete the following steps: **1.** Reset the interface card. **2.** Check the cables. **3.** If the problem persists, contact your technical support.
209	Excessive unsupported protocol packets are being sent to your storage system.	The problem is not with your storage system. Contact your network administrator to resolve the problem.
301	The IP address and the netmask are inconsistent with the assigned broadcast address.	Change the configuration by using the `ifconfig` command.
302	The broadcast address reaches a larger set of hosts than the standard broadcast computed from the IP address and netmask.	If this behavior is erroneous, change the configuration.
303	There are excessive IP reassembly errors.	Switch from NFS over UDP to NFS over TCP.

Error code	Description	Recommended actions
401	The TCP window advertised by the client is too small.	The problem is not with your storage system. Reconfigure the client.
402	There is excessive packet loss on the sending side.	The problem is not with your storage system. Examine the network and the client for congestion.
403	There is excessive packet loss on the receiving side.	The problem is not with your storage system. Examine the network and the client for congestion.
404	The average TCP packet size is poor on the receiving side because the network, client, or both are not enabled to support jumbo frames.	The problem is not with your storage system. Enable support for jumbo frames in network devices and the client.
405	The average TCP packet size is poor on the receiving side because of a problem with the network, client, or both.	The problem is not with your storage system. Examine the network and client for configured MTUs.
406	The average TCP packet size is poor on the receiving side because of a client application problem.	The problem is not with your storage system. Examine the client application data transmission strategy.
407	Excessive TCP listen socket drops because the system is overloaded or under security attack.	Contact your network administrator to resolve the problem.
408	There are excessive filtered TCP port drops because the system is under security attack.	Check your network. Contact your network administrator to resolve the problem.

Error code	Description	Recommended actions
409	There are excessive embryonic TCP connection drops because the system is under security attack or because a client has a bug.	A packet trace might assist in locating the problem. Contact your network administrator to resolve the problem.
410	Excessive TCP checksum errors. These errors can be caused by bad hardware on the client, in the network infrastructure (for example, blade in switch or router), or on the NIC. These errors can also be caused by a bug in the client.	• Check your client system for bugs. • Replace hardware components until the problem is resolved. • Contact your network administrator to resolve the problem.
411	There are packets because of a client. Your system might be under a security attack.	The problem is not with your storage system. • Check your client system for bugs. • Check for a security attack.
451	There are excessive UDP checksum errors.	Switch from NFS over UDP to NFS over TCP.
601	The DNS server is not reachable.	Examine the DNS server and the path to the DNS server.
602	The NIS server is not reachable.	Examine the NIS server and the path to the NIS server.

Copyright information

Trademark information

NetApp, the NetApp logo, Network Appliance, the Network Appliance logo, Akorri, ApplianceWatch, ASUP, AutoSupport, BalancePoint, BalancePoint Predictor, Bycast, Campaign Express, ComplianceClock, Cryptainer, CryptoShred, Data ONTAP, DataFabric, DataFort, Decru, Decru DataFort, DenseStak, Engenio, Engenio logo, E-Stack, FAServer, FastStak, FilerView, FlexCache, FlexClone, FlexPod, FlexScale, FlexShare, FlexSuite, FlexVol, FPolicy, GetSuccessful, gFiler, Go further, faster, Imagine Virtually Anything, Lifetime Key Management, LockVault, Manage ONTAP, MetroCluster, MultiStore, NearStore, NetCache, NOW (NetApp on the Web), Onaro, OnCommand, ONTAPI, OpenKey, PerformanceStak, RAID-DP, ReplicatorX, SANscreen, SANshare, SANtricity, SecureAdmin, SecureShare, Select, Service Builder, Shadow Tape, Simplicity, Simulate ONTAP, SnapCopy, SnapDirector, SnapDrive, SnapFilter, SnapLock, SnapManager, SnapMigrator, SnapMirror, SnapMover, SnapProtect, SnapRestore, Snapshot, SnapSuite, SnapValidator, SnapVault, StorageGRID, StoreVault, the StoreVault logo, SyncMirror, Tech OnTap, The evolution of storage, Topio, vFiler, VFM, Virtual File Manager, VPolicy, WAFL, Web Filer, and XBB are trademarks or registered trademarks of NetApp, Inc. in the United States, other countries, or both.

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. A complete and current list of other IBM trademarks is available on the web at *www.ibm.com/legal/copytrade.shtml*.

Apple is a registered trademark and QuickTime is a trademark of Apple, Inc. in the United States and/or other countries. Microsoft is a registered trademark and Windows Media is a trademark of Microsoft Corporation in the United States and/or other countries. RealAudio, RealNetworks, RealPlayer, RealSystem, RealText, and RealVideo are registered trademarks and RealMedia, RealProxy, and SureStream are trademarks of RealNetworks, Inc. in the United States and/or other countries.

All other brands or products are trademarks or registered trademarks of their respective holders and should be treated as such.

NetApp, Inc. is a licensee of the CompactFlash and CF Logo trademarks.

NetApp, Inc. NetCache is certified RealSystem compatible.

How to send your comments

You can help us to improve the quality of our documentation by sending us your feedback.

Your feedback is important in helping us to provide the most accurate and high-quality information. If you have suggestions for improving this document, send us your comments by email to *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.

You can also contact us in the following ways:

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

Index

/etc/gateways file 55
/etc/hosts file
    about 64
    adding, host name 65
    changing host name 66
    creating, from NIS master 75
    hard limits 66
    host-name resolution 64
/etc/nsswitch.conf file 64
/etc/resolv.conf file 67, 69, 70
/etc/resolv.conf file, hard limits 68
/etc/services file 173, 177
/etc/syslog.conf file 184

10 Gigabit Ethernet controller
    LINK INFO statistics 161–163
    RECEIVE statistics 161–163
    statistics 161–163
    TRANSMIT statistics 161–163
10 Gigabit Ethernet interface
    statistics 154, 156

A

A record 68
AAAA record 68
ABC
    ABC
        about 148
    improving TCP network congestion with 148
access
    restricting protocol 37
ACP
    defined 18
    enabling 18
address autoconfiguration 48, 49
address resolution 49
alias address
    creating 34
    deleting 34
Alternate Control Path (ACP)
    defined 18
anycast address 45

B

Baseboard Management Controller (BMC) 14
Baseboard Management Controllers
    See BMCs
blocking protocols 35
BMCs
    ways to configure 15

C

CDP
    configuring hold time 122
    configuring periodicity 123
    Data ONTAP support 121
    disabling 122
    enabling 122
    online migration 121
    viewing neighbor information 125
    viewing statistics 123
CDP (Cisco Discovery Protocol) 121
CIFS (Common Internet File Service) 182
Cisco Discovery Protocol (CDP) 121
commands
    dns flush 69
    dns info 69
    ifconfig 23, 24, 28, 34, 38, 56, 89
    ifconfig -a 38
    ifgrp add 109
    ifgrp create 101
    ifgrp create lacp 107
    ifgrp create multi 106
    ifgrp create single 102
    ifgrp delete 110
    ifgrp destroy 114
    ifgrp favor 104
    ifgrp nofavor 105
    ifgrp stat 113
    ifgrp status 111
    ifstat 28, 38, 40, 154, 156, 167–169
    ndp 143
    netdiag 143, 144, 186
    netdiag -s 144
    netstat 38, 39
    netstat -p icmp 146
    netstat -rn 59, 60
    nis info 79, 80
    ping 143
    ping6 143

pktt 143
route 54, 56, 61
route -s 59
route add 56
routed 58
routed status 61
snmp 131, 132
snmp authtrap 132
snmp community 132
snmp contact 132
snmp init 132
snmp location 132
snmp traphost 132
snmp traps 127, 132, 136–138
snmpbulkget 134, 135
snmpbulkwalk 134, 135
snmpwalk 130, 134, 135
sysconfig 12
traceroute6 143
useradmin group add 130
useradmin role add 130
useradmin user add 130
vlan add 87, 91
vlan create 87, 88
vlan delete 87, 92
vlan modify 87, 92
vlan stat 87, 93, 94
Common Internet File Service (CIFS) 182
congestion window (cwnd)
increasing the upper limit 149

D

DAD (Duplicate Address Detection) 50
default route
e0M 57
default router list 56
DHCP 180
diagnose network problems 143
diagnostic tests 143, 144
DNS
about 66
configuration information 69
configuring, from the command-line interface 67
disabling 67
disabling, dynamic updates 71
dynamic updates 70, 71
enabling 67
enabling, dynamic updates 71
fully qualified domain names (FQDN) 69
host-name resolution 64, 66, 68
lookup 68
modifying dns.update.ttl 73
name cache 69
port used 179
time-to-live (TTL) 70
Domain Name System (DNS) 66
Dual 10G Ethernet controller
LINK INFO statistics 164–166
RECEIVE statistics 164–166
statistics 164–166
TRANSMIT statistics 164–166
duplex settings, correcting mismatches 171
Duplicate Address Detection (DAD) 49, 50
dynamic DNS
about 70
disabling 71
disabling, for an IP address 72
enabling 71
in Data ONTAP 71
Dynamic Host Configuration Protocol (DHCP) 66, 180

E

e0M
blocking data traffic 36
guidelines for configuring 14
subnet mask 25
e0M interface
introduction to 12
error messages
error code, netdiag 186
networking 186
Ethernet frame
jumbo frame 22

F

failover, second-level 116
fast path
about 53
differences between IPv4 and IPv6 54
disabling 57, 58
enabling 57, 58
IPv4 54
IPv6 54
similarities between IPv4 and IPv6 54
TSO 58
with asymmetric routing 53
with NFS-over-UDP 53

with TCP 53
See also fastpath
fastpath 53, 58
File Transfer Protocol (FTP) 177
flow control
about 22, 28
options 28
frame
about 21
characteristics 21
Ethernet 21
flow 22
frame size 21
jumbo 21
jumbo frame 21
MTU size 21
Pause Off 22
Pause On 22
FTP 177
fully qualified domain names (FQDN) 69

G

GARP (Generic Attribute Registration Protocol) 84
GARP VLAN Registration Protocol (GVRP) 84
Generic Attribute Registration Protocol (GARP) 84
Gigabit Ethernet controller
LINK INFO statistics 150, 152, 153
RECEIVE statistics 150, 152, 153
statistics 150, 152, 153
TRANSMIT statistics 150, 152, 153
Gigabit Ethernet interface
interface statistics 154, 156
statistics 157, 159, 160
GVRP 86
GVRP (GARP VLAN Registration Protocol) 84

H

host
identification 173
naming 10, 11
host name
about 10, 11
adding, in /etc/hosts file 65
changing 66
resolution, with /etc/hosts file 64
resolution, with DNS 66
resolution, with NIS 73, 76
host-name resolution
about 64
using /etc/hosts file 64
using DNS 66, 68
using NIS 73, 76
HTTP 181
Hypertext Transport Protocol (HTTP) 181

I

ICMP 146, 147
ICMP redirect messages 147
ICMP Router Discovery Protocol (IRDP) 55
IEEE 802.1Q standards 86
ifgrp
creating single-mode 102
deleting interfaces 110
destroying 114
viewing statistics 113
inter-switch link (ISL) 115
interface
e0M, introduction 12
interface group
about 95
adding interfaces 109
dynamic multimode 96, 98, 107
dynamic multimode, LACP log 107
in an HA pair 117
LACP 107
load balancing 100
load balancing, IP address based 100
load balancing, MAC address based 100
manage 101
naming 10, 11
second-level 114, 115, 117, 119
selecting preferred interface 104
single-mode 96
specifying nonfavored interface 105
static multimode 96, 97, 106
status 111
types 96
viewing status 111
interface groups
single-mode, failure scenarios 106
IP address
alias 34
broadcast 26
configuration 23
configuring 24
flow control 28
media type 27

MTU size 27
partner IP, specifying 30
prefix length 26
removing 32
subnet mask 25
IP ports 173
IPv6
address autoconfiguration 48
address scopes 45
address states 46
address types 45
configure addresses 45
disabling 47
dual stack mechanism 46
dynamic routing 56
enabling 47
Router Advertisement 56
stateless address autoconfiguration 48
support in Data ONTAP 45

J

jumbo frames
advantages 21, 22
configuring 22
network requirements 22
size 21

K

Kerberos 181

L

LACP (Link Aggregation Control Protocol) 98
LACP log file 107
Large Receive Offload
See LRO
Link Aggregation Control Protocol (LACP) 98
LINK INFO statistics
10 Gigabit Ethernet interface 154, 156
Gigabit Ethernet interface 154, 156, 157, 159, 160
load balancing
IP address based 100
MAC address based 100
multimode interface groups 100
round-robin 100
localhost 64
LRO
CPU efficiency 17
defined 17
viewing statistics 17

M

management interface 35
MIB
/etc/mib/iscsi.mib 128
/etc/mib/netapp.mib 128
custom mib 128
iSCSI MIB 128
monitoring
network connectivity 121
multicast address 45
multimode interface groups
load balancing, IP address based 100
load balancing, MAC address based 100
load balancing, port-based 100
load balancing, round-robin 100

N

NDMP (Network Data Management Protocol) 185
negotiated failover 31
Neighbor Discovery 49
Neighbor Solicitation 32
neighbor unreachability detection 49
network connectivity
discovering 121
Network Data Management Protocol (NDMP) 185
Network File System (NFS) 181
Network Information Service (NIS) 73
network interface
10 Gigabit Ethernet 10
10/100/1000 Ethernet 10
100 Mbps 10
100BT 10
automatic takeover 31
blocking protocols 35
changing status 34
clearing statistics 40
configuration 23
configuring 23
dad_attempts 32
down, status 34
flow control 28
Gigabit Ethernet 10
maximum number 12
naming 10, 11

nfo 31
statistics 150
statistics for T204E 154, 156
statistics for T204V 154, 156
statistics for T210 154, 156
statistics for T320 154, 156
trusted 29
types 10
untrusted 29
up, status 34
viewing context statistics 39
viewing settings 38
viewing statistics 40
network interfaces
viewing statistics 38
Network Time Protocol (NTP) 179
next-hop determination 49
NFS
port used 181
NIC
CPU efficiency 15
LSO 15
TSO 15
NIS
about 73
administrative commands
ypcat 76
ypgroup 76
ypmatch 76
yppush 74
ypwhich 76
configure 76
creating /etc/hosts file 75
disabling 77
enabling 77
enabling slave 79
host-name resolution 64, 73, 76
hosts map 73, 74
ipnodes map 73, 74
IPv6 support 73
master 74, 75
netgroup cache 79
selecting the master server 74
slave 74, 75
specifying domain name 77
specifying servers 78
statistics 79
viewing information 79
viewing, performance statistics 80
NIS (Network Information Service) 73
NIS slave
about 74
enabling 79
guidelines 75
improve performance 74
NNTP 176
NTP 179

O

OID 128
options
dns.cache.enable 69
dns.update.enable 71
dns.update.ttl 73
interface.blocked.CIFS 35
interface.blocked.FTP 35
interface.blocked.iSCSI 35
interface.blocked.NDMP 35
interface.blocked.NFS 35
interface.blocked.SnapMirror 35
interface.blocked.SnapVault 35
ip.fastpath.enable 58
ip.icmp_ignore_redirect.enable 56, 147
ip.ping_throttle.drop_level 145, 146
ip.tcp.abc.l_limit 149
ip.tcp.rfc3390 148
ip.v6.enable 47
ip.v6.ra_enable 48
nis.domainname 77
nis.enable 77
nis.server 75
nis.servers 74, 78
nis.slave.enable 79
snmp.access 131
snmp.enable 130

P

parameter discovery 49
pause frame 22
performance, storage system
e0M 171
ping
command 143
diagnose problems 145
throttling 145
throttling, disabling 146
throttling, threshold status 146
throttling, threshold value 146

port
    for SnapMirror 185
    for SnapVault 185
    NDMP 185
    NFS 181
port usage 173
ports
    TCP 176
    UDP 176
ports, IP 173
ports, NFS-enabled 176
prefix discovery 49
prefix list 56
protocol
    restricting access 37
protocol filter
    viewing statistics 43

Q

Quad Gigabit Ethernet Controller
    LINK INFO statistics 167–169
    TRANSMIT statistics 167–169
Quad Gigabit Ethernet interface
    RECEIVE statistics 167–169

R

RA messages 48
RECEIVE statistics
    10 Gigabit Ethernet interface 154, 156
    Gigabit Ethernet interface 154, 156, 157, 159, 160
redirect by routers 49
Remote LAN Module (RLM) 14
Remote LAN Modules
    See RLMs
Remote Shell (RSH) 184
reverse lookup 68
RLMsways to configure 15
route
    default 57
route metric 61
routed daemon
    about 55
    disable 57
    enable 57
    port usage 184
    turning off 55, 58
    turning on 58
Router Advertisement 56
router advertisement (RA) 48
router advertisement messages 48
router discovery 49
router-advertised messages
    disabling 48
    enabling 48
routing
    about 53
    default route 57, 59, 61
    fast path 57, 58
    managing routing table 54
    methods 53
    modifying routing table 61
    routed daemon 55, 57, 58
    routing table 56, 57, 59
    vFiler units 56
routing information 61
routing protocols 61
routing table
    commands to manage 54
    flags 60
    IPv6 56
    modify, circumstances 56
    modifying 61
    vFiler units 56
    viewing 59
RSH 184

S

SAS shelves
    ACP protocol 18
second-level interface group
    guidelines for creating 115
Secure Shell (SSH) 178
Secure Sockets Layer (SSL) 182
services file 173
Simple Mail Transport Protocol (SMTP) 179
Simple Network Management Protocol (SNMP) 127, 183
SMTP 179
SNMP
    access privileges, setting 131
    agent 127, 129
    agent, configure 129
    authKey security 133
    authNoPriv security 133
    authProtocol security 133
    cluster Vserver 127
    commands 132, 134, 135
    configuring group, v3 130

configuring role, v3 130
configuring users, v3 130
disabling 130
enabling 130
examples 134, 135
login-snmp capability, v3 130
MIBs 127, 128
modifying configuration 131
noAuthNoPriv security 133
port usage 183
restricting access 131
security parameters 133
traps 128
traps, define 136
traps, examples 137
traps, guidelines for creating 136
traps, modify 136
traps, modifying 137
traps, parameter 140
traps, parameters 138–141
traps, types 127
traps, user-defined 136
traps, viewing 137
viewing configuration 131
SNMP (Simple Network Management Protocol) 127
SNMP traps
backoff-calculator parameter 140
backoff-multiplier parameter 141
backoff-step parameter 141
built-in 127
commands 138
creating 136
edge-1 parameter 139
edge-1-direction parameter 140
edge-2 parameter 139
edge-2-direction parameter 140
example 137
guidelines 136
interval parameter 140
interval-offset parameter 140
message parameter 141
modifying 136, 137
parameters 138
priority parameter 141
rate-interval parameter 140
standard 127
trigger parameter 139
user-defined 127, 136
var parameter 139
viewing 137
SNMPv3
configuring group 130
configuring role 130
configuring users 130
example 134, 135
login-snmp capability 130
split-network condition 115
SSH 178
SSL 182
statistics
Gigabit Ethernet interface 157, 159, 160
viewing LRO 17
storage system 127
syslog 184

T

TCP Segmentation Offload (TSO) 16
Telnet 178
TFTP 180
time service 179
time-to-live (TTL) 70, 73
TLS 182
TRANSMIT statistics
10 Gigabit Ethernet interface 154, 156
Gigabit Ethernet interface 154, 156, 157, 159, 160
Transport Layer Security version (TLS) 182
transport layer, diagnosing 144
Trivial File Transfer Protocol (TFTP) 180
TSO
Data ONTAP support 16
fast path 16
viewing statistics 16
TSO (TCP Segmentation Offload) 16
TTCP 176

U

unicast address 45

V

vfiler
configuring RA prefix 48
VLAN
adding an interface 91
commands 87
configuring 86, 89
configuring GVRP 84

creating 88
deleting 92
GVRP 84
link-local address 90
membership 83
modifying 92
naming 10, 11
prerequisites 86
tags 84
viewing statistics 93, 94
VLANs
advantages 85
tagged traffic 90
tagging 82
untagged traffic 90