Technical Report

Ethernet Storage Design Considerations and Best Practices for Clustered Data ONTAP Configurations

Kris Lippe, NetApp
January 2016 | TR-4182-0216

Abstract

This technical report describes the implementation of NetApp® clustered Data ONTAP® network configurations. It provides common clustered Data ONTAP network deployment scenarios and recommends networking best practices as they pertain to a clustered Data ONTAP environment. A thorough understanding of the networking components of a clustered Data ONTAP environment is vital to successful implementations.

This report should be used as a reference guide only. It is not a replacement for product documentation, specific clustered Data ONTAP technical reports, end-to-end clustered Data ONTAP operational recommendations, or cluster planning guides. In addition, any solution-specific guides will supersede the information that is contained in this document, so you should double-check all related references.

Data Classification

Public, NetApp internal and external.

Version History

Version	Date	Document Version History
Version 1.2	January 2016	Added troubleshooting section for Path MTU black-hole detection and updated guidance for flow control settings
Version 1.1	June 2015	Updates for 8.3.1, which include enhancements to IPspaces, broadcast domains, and SVM management
Version 1.0	March 2015	Mike Worthen: Initial commit

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 Overview

With NetApp clustered Data ONTAP 8.2 networking:

- We introduced support for IPv6 in the clustered Data ONTAP codeline.
- Failover groups became easier to configure correctly because more of the UI parameters were written into the functionality of clustered Data ONTAP. Specifically, fewer parameters need to be configured by the user now. Clustered Data ONTAP handles some of what needed to be specified manually in earlier clustered Data ONTAP versions.
- Virtual LANs (VLANs) are supported if DNS load balancing is configured.

With clustered Data ONTAP 8.3 networking:

- IPspaces are introduced into the clustered Data ONTAP codeline. This change now allows the use of custom IPspaces, the default IPspace, and overlapping IP addresses. Broadcast domains (layer 2) are introduced into the clustered Data ONTAP codeline and are now the objects that define failover groups in clustered Data ONTAP.
- Beginning with clustered Data ONTAP 8.3, node management and cluster management logical interfaces (LIFs) can no longer be used by a storage virtual machine (SVM) to make connections to outside resources such as Active Directory or DNS. With clustered Data ONTAP 8.3, there is a better definition of uniform network access between the SVMs and the nodes in a cluster. This new definition is described in section 3.8.
- Routing changes have occurred between clustered Data ONTAP 8.2.x and clustered Data ONTAP 8.3.x.
- As the adoption of IPv6 has increased in NetApp's customer base, NetApp has increased its support and innovation for IPv6 in the clustered Data ONTAP codeline.

We have now covered the overview of clustered Data ONTAP networking changes past and present. So, let's move on to discuss the different types of cluster configurations that can be implemented, all of which use various networking concepts and features.

1.1 Cluster Choices: Types of Clusters Currently Available

- **Single-node cluster (Figure 1).** In a single-node cluster, settings such as Ethernet flow control and TCP options still need to be properly configured. Also, if you plan to upgrade the configuration from a single node, NetApp recommends as a best practice that you install the necessary components for the upgrade and expansion during the initial implementation. For example, you should install network interface cards (NICs) for cluster interconnectivity. This step could save a reboot or two when the need to move to a highly available solution presents itself.

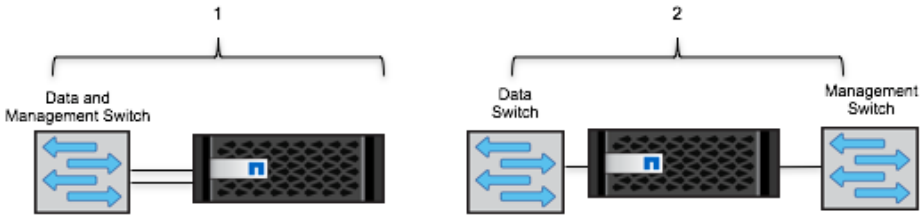
Note: In a single-node configuration, there aren't other nodes in the cluster to which you can replicate important cluster and node configuration information. Therefore, you must make accommodations so that a backup copy is located off of the single node itself. For more information, see the section "Backing up and restoring cluster configurations" in the "System Administration Guide for Cluster Administrators" on support.netapp.com.

- **Two-node switchless cluster (Figure 2).** In a two-node switchless cluster configuration, settings such as flow control, TCP options, and maximum transmission unit (MTU) size also need to be configured appropriately. If there is a chance that you will move this configuration to a multinode switched configuration later, keep in mind that you can move it nondisruptively.

Note: The lack of a switch in this configuration does not change the port requirements per platform. See the Hardware Universe for port requirements and follow the same guidance whether it's a switched or switchless configuration.

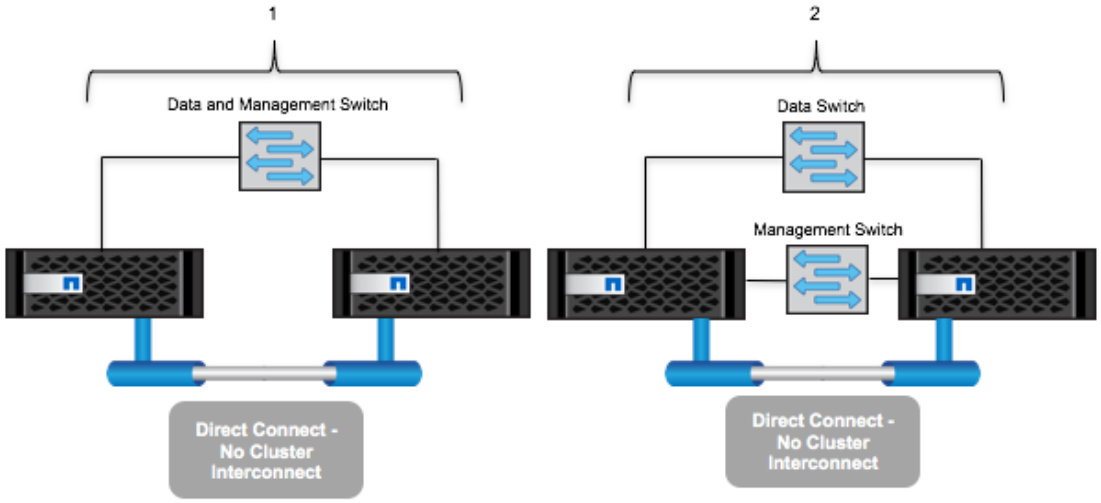
- **Multinode switched cluster (Figure 3).** In a multinode switched cluster, the customer gains all the benefits that clustered Data ONTAP offers: the nondisruptive capabilities, the highly available capabilities, and the performance capabilities. Although this solution is the most complex of the three to configure, it is also the one with the greatest return.

Figure 1) Single-node cluster.



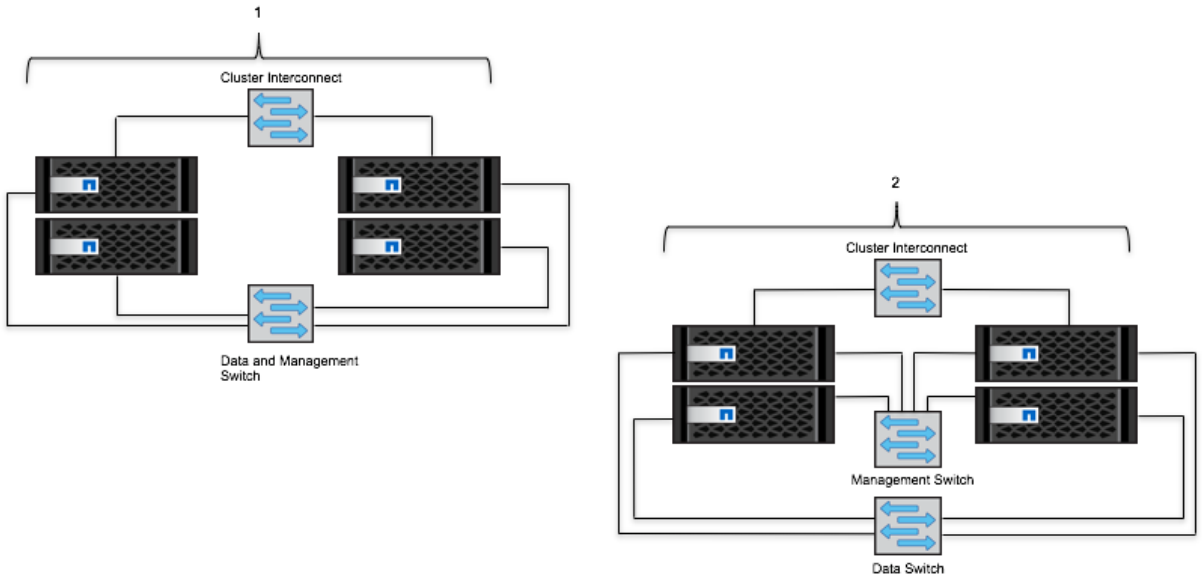
Note: Two options are available. In option 1, each cluster type has a switch configuration in which data and management requests are sent and received on the same switch infrastructure. In option 2, the configuration uses dedicated switches, one for data requests and one for management requests.

Figure 2) Two-node switchless cluster.



Note: Two options are available. In option 1, each cluster type has a switch configuration in which data and management requests are sent and received on the same switch infrastructure. In option 2, the configuration uses dedicated switches, one for data requests and one for management requests.

Figure 3) Multinode switched cluster.



Note: Two options are available. In option 1, each cluster type has a switch configuration in which data and management requests are sent and received on the same switch infrastructure. In option 2, the configuration uses dedicated switches, one for data requests and one for management requests.

2 Requirements

2.1 Setting Up the Cluster

Several software and hardware prerequisites must be met in initially setting up and configuring a NetApp clustered Data ONTAP implementation. We begin with discussing the port types that are available. Then we go through the logical interfaces (LIFs) that use the different port types to send and receive I/O requests to and from the cluster:

- Ports:
  - Physical ports: These ports are used for various functions and can have different types of configurations.
  - Logical ports: Virtual LANs (VLANs)/(802.1Q) and interface groups (IFGRPs) make up the options for logical ports.

Note: Port roles have been deprecated beginning with clustered Data ONTAP 8.3. IPspaces and broadcast domains allow the administrator to enforce the rule that certain ports be reserved exclusively for specific LIFs.

- LIFs:
  - Data
  - Cluster
  - Cluster management
  - Node management
  - Intercluster
  - Storage virtual machine (SVM) management

3 Architecture

3.1 IPspaces: Past and Present

Beginning in NetApp clustered Data ONTAP 8.3, users have the ability to create unique IPspaces (or they might choose to use only the default IPspace). The IPspace feature enables a single storage cluster to be accessed by clients from more than one disconnected network, even if those networks have overlapping IP addresses. The IPspace object is not new to Data ONTAP and clustered Data ONTAP; there are similarities and differences with IPspaces in clustered Data ONTAP 8.3. One significant difference to keep in mind with IPspaces in clustered Data ONTAP 8.3 versus Data ONTAP operating in 7-Mode is that in 7-Mode, each vif shared a routing table. But with clustered Data ONTAP 8.3, each SVM maintains its own routing table (see Figure 4 and Figure 5).

Note: In the networking world, IPspaces are equivalent to Virtual Routing and Forwarding, a technology that allows multiple instances of a routing table to coexist on the same piece of networking infrastructure.

Figure 4) IPspaces in clustered Data ONTAP 8.3: routing unique to each SVM.

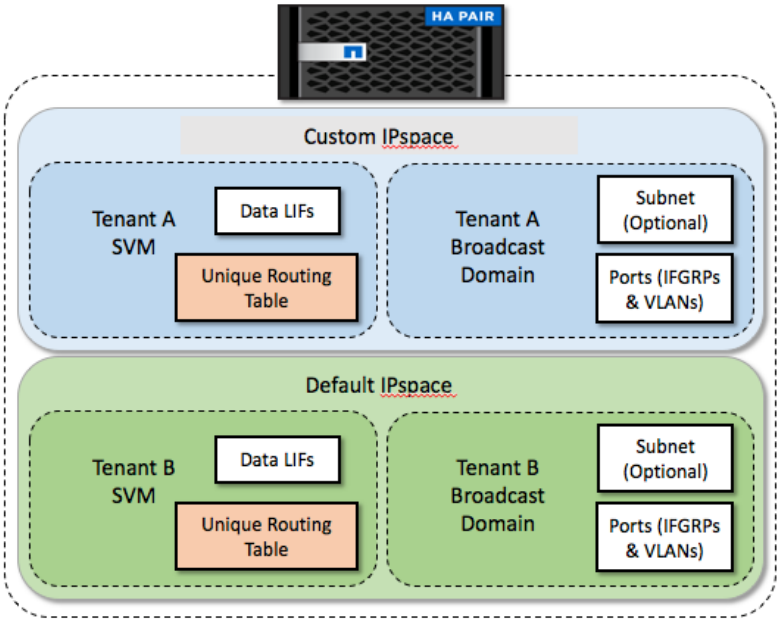
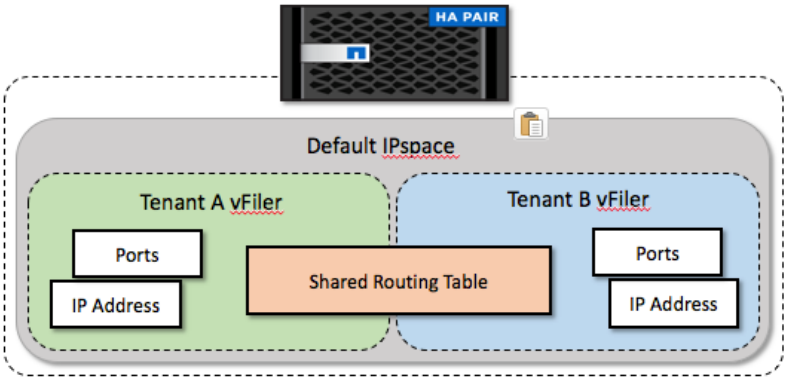


Figure 5) IPspaces in 7-Mode: shared routing table.



IPspace features include the following:

- Two hosts in two different IPspaces can have the same IP address and not conflict with each other. This feature is useful if, for example, different entities are being collapsed onto the same infrastructure. The transition is made simpler if you do not have to ask the host-side administrators or developers to change IP addresses or host-name references.
- The port selection options for different IPspaces are made up of physical ports, IFGRPs, and VLANs. However, port roles have been deprecated and are no longer a consideration.
- An IPspace provides a logical construct that owns the SVMs that reside in it. The SVMs are their own secure objects within an IPspace.
- The number of IPspaces required: As mentioned, users aren't required to create additional IPspaces but might choose instead to use the default IPspace.

  Reasons that the default IPspace might be the best choice include:

  – **Express setups.** You can quickly deploy proof-of-concept-type scenarios.
  – **Remote offices.** A single-node cluster is needed for a backup source.
  – **The easiest and most effective choice.** There are no requirements dictating that the default IPspace cannot be used, and with earlier versions of clustered Data ONTAP, the default IPspace was the only one used. This option balances administration with complexity.

  Reasons that more than the default IPspace might be needed include:

  – **Business units.** For example, if the Legal department of a company needs a repository, it can be confident that the repository is isolated from the rest of the company's data.
  – **Hosted services in which different tenants might exist on the same cluster.** However, this segregation can also be achieved by having the different tenants in different SVMs in the same IPspace. An SVM that is configured correctly is a secure container, and it's secure whether it's in its own IPspace or it's sharing one with other SVMs and tenants.

IPspace considerations include the following:

- **SVM relocation.** After an SVM is created, it cannot be moved to another IPspace in 8.3.x. You must re-create the SVM in a different IPspace and you must port any data or settings to the new SVM. This consideration is critical if you upgrade from an earlier clustered Data ONTAP version to clustered Data ONTAP 8.3. Any existing SVMs are relocated to the default IPspace and cannot be moved. For more information, see section 9.4 in the "Use Cases" section.
- **Cluster peering configurations.** Starting with clustered Data ONTAP 8.3.1, the default IPspace is no longer the only IPspace that can host intercluster LIFs; custom IPspaces can also be used to host these interfaces. Furthermore, the intercluster LIFs do not need to reside in the same IPspace as the replicating SVM. This flexibility means that an SVM in one IPspace can leverage the intercluster LIFs created in another IPspace. See Figure 10 for more details.

3.2 Broadcast Domains

A broadcast domain in clustered Data ONTAP is a layer 2 object that allows the logical division of resources in a clustered Data ONTAP system. This feature occurs at the port layer; physical and logical ports that make up a layer 2 network are combined into Data ONTAP objects called *broadcast domains*. The ports in the broadcast domain can then be used by LIFs within an SVM to host data or management traffic. To put it simply, a broadcast domain can be thought of as the Data ONTAP interpretation of the underlying physical network, and it should be configured as such.

When discussing broadcast domains, it's worth mentioning two other clustered Data ONTAP network constructs, namely:

- **Failover groups.** A broadcast domain controls the creation of them, but it shares "ownership" of a failover group with an SVM.
- **Subnets.** Subnets can be used to assign IP addresses automatically; it is optional based on the needs of the configuration.

In the case of clustered Data ONTAP, all the layer 2 broadcast traffic is scoped to an IPspace because broadcast domains in clustered Data ONTAP cannot span IPspaces.

Broadcast domains share the following features:

- A failover group is created automatically each time a broadcast domain is created, and the name of the failover group is set as the same as the broadcast domain:
  - As physical or logical ports are added to or removed from the broadcast domain, the same operation is performed on the associated failover group. As a result, the ports in the broadcast domain mirror the ports of the failover group.
- They allow definition of the networks to which the cluster is connected.
- They help enforce rules that apply to existing objects such as ports, failover groups, and interfaces.
- They lay the groundwork to perform broadcast domain autodetection and verification.

Considerations and requirements include the following:

- All ports that are used in the same layer 2 broadcast domain must have the same MTU value (clustered Data ONTAP enforces this requirement, so the only way it can be changed is manually).
- All ports in failover groups must be in the same layer 2 broadcast domain.
- The home port, current port, failover group, and subnet object of an interface must have the same broadcast domain.
- It should be verified that LIFs do not fail over to ports that aren't connected to the same network.

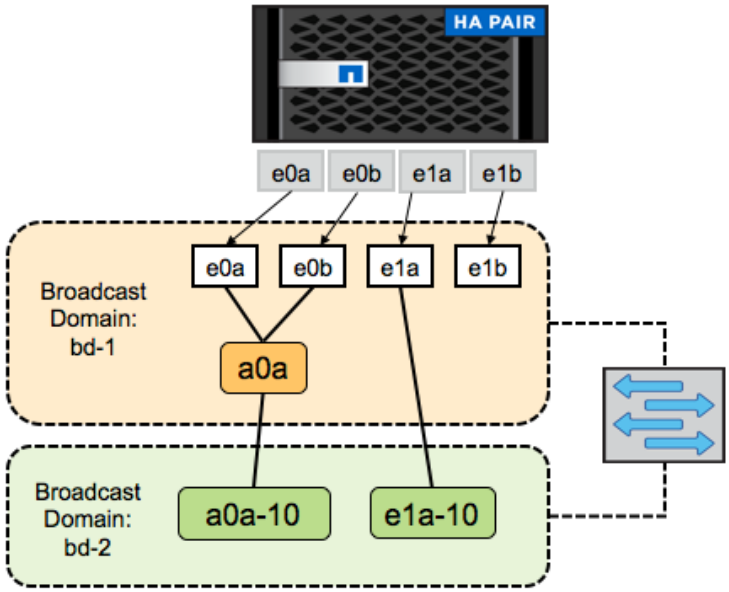
Note: Broadcast domains are composed of physical ports, VLAN tagged logical ports, and interface groups (IFGRPs). Although a broadcast domain can contain any mix of these three port types, a port can be assigned to only one broadcast domain.

Figure 6 shows an example in which broadcast domains are used to segregate traffic across multiple VLANs.

In this example, the Data ONTAP cluster is configured with four physical ports, `e0a`, `e0b`, `e1a`, `e1b`. Two broadcast domains are configured, with all physical untagged ports grouped into broadcast domain `bd-1` and a set of VLAN tagged ports (`VLAN 10`) grouped into broadcast domain `bd-2`. As explained previously, broadcast domains create and manage the port membership for a "default" associated failover group. Therefore, we place VLAN tagged ports into one domain and untagged ports into another. With this approach, in the event of a LIF failover, as long as the LIF subscribes to the broadcast domain failover group, the destination port is part of the same L2 network.

By allowing broadcast domains to manage the port membership of the associated failover group, we make network changes less error-prone for the storage administrator. When changes are made to the underlying network configuration, by updating the broadcast domain appropriately, we can be confident that our LIFs will fail over to the correct ports. This approach prevents LIF outages that can occur when a system administrator forgets to update the failover group ports.

Figure 6) Introduction to broadcast domains in clustered Data ONTAP 8.3.



3.3 Clustered Data ONTAP Port Types

There are different types of ports in clustered Data ONTAP: physical and logical. The different port types are used throughout the cluster in different configurations but are the building blocks that logical interfaces (LIFs) use to allow the sending and receiving of data.

Physical Ports

Physical ports can be used individually or in combination when configuring logical ports. If you use several physical ports together for an IFGRP, it is important to remember to configure all relevant port settings in the same way (including MTU size and flow control). However, these settings are also relevant if you exclusively use physical ports in a configuration in which only failover groups are in play. Consistency in the settings and in the configurations is a recommended best practice.

VLANs

A VLAN subdivides a physical network into distinct broadcast domains. As a result, traffic is completely isolated between VLANs unless a router (layer 3) is used to connect the networks. From a security perspective, complete isolation is one of the primary reasons to use VLANs. In clustered Data ONTAP, VLANs subdivide a physical port into several separate logical ports, allowing one of the key components of our secure multi-tenant messaging, which is the isolation of data.

IFGRPs

IFGRPs can be configured to add another layer of redundancy and functionality to a clustered Data ONTAP environment. They can also be used in conjunction with a failover group, which helps protect against layer 2 and layer 3 Ethernet failures. An IFGRP can somewhat be compared with a vif from earlier Data ONTAP and clustered Data ONTAP versions. However, there is no longer a need to group multiple multimode vifs or IFGRPs to obtain a second-level vif or IFGRP. The layer of redundancy and aggregated performance that this configuration gave is now achievable by using a dynamic multimode IFGRP:

- **With clustered Data ONTAP 8.2.x.** IFGRP physical member ports must all be part of the same subnet. If physical member ports from different subnets are mistakenly configured together in the same IFGRP port, connectivity issues arise, and loss of data connectivity occurs. After an IFGRP is

configured correctly with physical member ports from the same subnet, it is then part of the overall network configuration of the cluster. This attribute is slightly different with clustered Data ONTAP 8.3.x.

- **With clustered Data ONTAP 8.3.x.** IFGRP ports can be part of a broadcast domain if they contain physical member ports that belong to the broadcast domain. An IFGRP without any physical member ports cannot host LIFs, be in a failover group, or be part of a broadcast domain. Removing the last physical member port from an IFGRP causes the IFGRP to be removed from the broadcast domain.

Best Practice: IFGRP Resiliency

As a best practice, NetApp recommends that when you create an IFGRP, you should create the IFGRP by using ports from different NICs. You should, however, verify that they are the same model or chipset and that they have the same port speed, functionality, and so on. This step is critical in maintaining consistency in the IFGRP in the event of a port failure. By maintaining consistency with port aggregation and by spreading the IFGRP over NICs in different PCI slots, you decrease the chances of a slot failure bringing down all the ports in an IFGRP.

Limitations in either network connectivity or port availability might prevent the administrator from adding ports from multiple NICs into the same IFGRP, as explained in the preceding best practice. In that case, to prevent an ASIC failure from taking down multiple ports within an IFGRP, NetApp recommends that IFGRPs use ports that do **not** share the same ASIC.

For example, let's say a NIC within your Data ONTAP cluster contains four physical ports. Let's also say that ports `e0a` and `e0b` share one ASIC, and ports `e0c` and `e0d` share another. In this case, the best practice is to place ports `e0a` and `e0c` into one IFGRP and to place ports `e0b` and `e0d` in a second IFGRP.

Note: If you use iSCSI (RFC 3720) in clustered Data ONTAP, there are two methods to achieve path redundancy. One method is to use IFGRPs to aggregate more than one physical port in partnership with a Link Aggregation Control Protocol (LACP)–enabled switch. Another method is to configure hosts to use Microsoft Multipath I/O (MPIO) over multiple distinct physical links.

Both of these methods allow a storage controller and a host to use aggregated bandwidth, and both can survive the failure of one of the paths from host to storage controller. However, MPIO is already a requirement for using block storage with clustered Data ONTAP, and the use of MPIO has the further advantage of no additional required switch configuration or port trunking configuration. Also, the use of an IFGRP for path management when you use iSCSI is not supported; again, you should use MPIO with iSCSI. However, the use of an IFGRP as a port for an iSCSI LIF is supported.

The following provides characteristics for each IFGRP type, as well as situations in which it is advisable to use each type.

- **Single-mode**

  A single-mode IFGRP is an active-passive configuration (one port sits idly waiting for the active port to fail), and it cannot aggregate bandwidth. Other than basic connectivity, the switch configuration on a single-mode IFGRP is not continuously verified (speed, duplex). As a result, if the VLAN connectivity that encompasses both ports is severed (for example, that VLAN is not configured on an alternate switch-to-switch trunk line becoming active), unexpected behavior might result. That behavior might occur in particular during reboots or cluster failover (either the "wrong" port becomes active or no port becomes active). Trying to enable the proper port works only if the node port was fully operational and in the proper state to begin with.

  Because of this IFGRP's limited capabilities, NetApp recommends as a best practice not using this IFGRP type. To achieve the same level of redundancy, you could use failover groups or one of the other two following IFGRPs.

- **Static multimode**

If you want to use all the ports in the group to simultaneously service connections, you can employ a static multimode (802.3ad) IFGRP. It does differ from the type of aggregation that occurs in a dynamic multimode IFGRP in that no negotiation or autodetection occurs within the group in regard to the ports. A port sends data when the node detects a link regardless of the state of the connecting port on the switch side. This issue can be very serious, because it can result in the IFGRP falling into a "black hole" scenario. In this scenario, the port continues to send packets, thinking that the link is still OK to receive, but it really isn't.

Note: For all data ports, and in particular for single-mode and static multimode IFGRPs, NetApp recommends one of two actions. You should either disable the spanning tree on the adjacent switch ports or set all the port-specific timers of the spanning tree to the minimum attainable. This action reduces any loss of service time due to the network's not properly accepting and forwarding storage data packets.

- **Dynamic multimode**

  A dynamic multimode IFGRP is sometimes called an LACP (802.1AX-2008) IFGRP because the Link Aggregation Group bundle works in conjunction with LACP. If you want additional error detection capabilities in the configuration, you might use this IFGRP to aggregate bandwidth of more than one port. LACP monitors the ports on an ongoing basis to determine the aggregation capability of the various ports. It also continuously provides the maximum level of aggregation capability that can be achieved between a given pair of devices.

  All the interfaces in the group are active, they share the same MAC address, and they handle load balancing of outbound traffic. But these combined attributes do not mean that a single host achieves larger bandwidth, exceeding the capabilities of any of the constituent connections. For example, adding four 10GbE ports to a dynamic multimode IFGRP does not result in one 40GbE link for one host. This limitation stems from the way in which the aggregation of the ports in the IFGRP is handled by both the switch and the node.

  As a best practice, NetApp recommends that you use this type of IFGRP so that you can take advantage of all the performance and resiliency functionality that the IFGRP algorithm has to offer.

Best Practice: IFGRP Port Settings
The network interfaces and the switch ports that are members of a dynamic multimode (LACP) IFGRP must be set to use the same speed, duplex, and flow control settings. However, NetApp also recommends that you follow the same practices if you create any of the following different IFGRP types.

Load Balancing for Multimode IFGRPs

Four distinct load-balancing modes are available:

- **Port.** Use this distribution method for optimal load-balancing results. However, it doesn't lend itself as well to troubleshooting, because the TCP/UDP port of a packet is also used to determine the physical port that is used to send a particular packet. It has also been reported that switches operating in particular modes (mapping MAC/IP/port) might exhibit lower than expected performance in this mode.
- **MAC.** This method is useful only when the IFGRP shares the same VLAN with the clients that have access to the storage. If any storage traffic traverses a router or a firewall, do not use this type of load balancing. Because the MAC address for every outgoing IP frame is the MAC address of the router, using this type of load balancing results in only one interface in the IFGRP being used.
- **IP.** This method is the second best for load distribution, because the IP addresses of both the sender (LIF) and the client are used to deterministically select the particular physical link that a packet traverses. Although it is deterministic in the selection of a port, the balancing is performed by using an advanced hash function. This approach has been found to work under a wide variety of circumstances, but particular selections of IP addresses might still lead to unequal load distribution.
- **Sequential.** This mode offers nondeterministic load balancing. Under specific circumstances, this type of load balancing can cause performance issues due to high overhead to the switch (potential

constant remapping of MAC/IP/port) or out-of-order delivery of individual packets destined for a client. Because of these potential issues, this type of load balancing is not supported by the IEEE LACP specification. And because of the potential for less than favorable load balancing and the potential for out-of-order delivery, NetApp does not recommend that you use this type.

Note: Remember, the load balancing in an IFGRP occurs on outbound traffic, not inbound traffic. So, when a response is being sent back to a requester, the load-balancing algorithm comes into play to determine which "path" is optimal to use to send the response back. Also, it's important to note that the preceding load-balancing options might differ from other settings in the environment. You should thoroughly check other devices that might be connected to the ports on the nodes in the cluster. If you use "port" on the IFGRP configuration on the cluster, make sure that the switch port on the Cisco, Juniper, Brocade, or other device is also configured in the same way.

3.4 Failover Groups

In the event of a failure, LIFs must be failed over to suitable targets in a coordinated manner. When a LIF is created, it is assigned to a system-defined failover group by default. However, the behavior of the system-defined failover group might not be sufficient for every different type of environment.

Clustered Data ONTAP 8.1.x to 8.2.x Releases

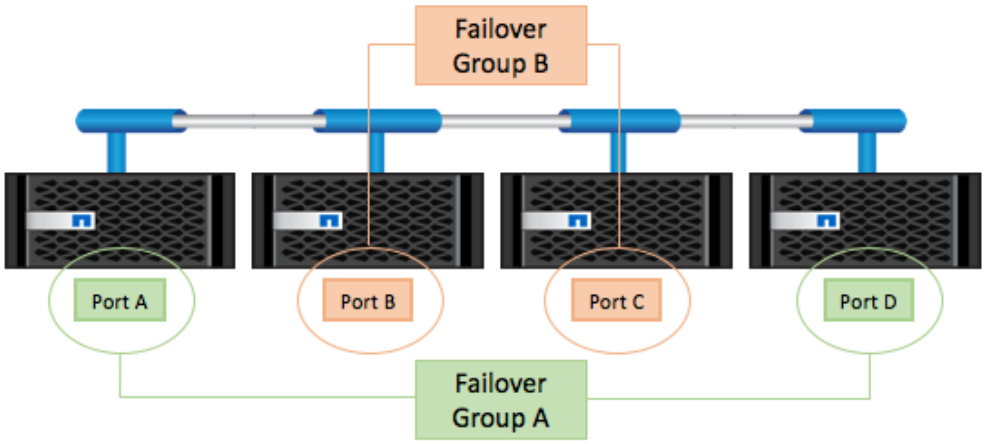
A failover group contains a set of network ports on one or more nodes. A failover group can have cluster management, node management, intercluster, and NAS data LIFs assigned to it. As mentioned previously in this document, SAN LIFs don't fail over, so they don't use failover groups. The network ports that are present in the failover group define the failover targets for the LIF. The recommended best practice for LIFs that are capable of using failover groups is to always assign those LIFs to an appropriate failover group. Also, it's extremely important to verify that all ports in the failover group are part of the same subnet. Failure to determine that ports are in the same subnet and failure to assign LIFs to appropriate failover groups can result in loss of connectivity to data.

There are currently three different types of failover groups. The following are descriptions of each type, with details about when you might want to use each:

- **User defined.** The recommended best practice is to use this type of failover group. Because of its flexible nature, you can configure it to provide both the redundancy and the performance that any environment requires. Some use cases to keep in mind when considering this type of failover group include:
  - If multiple subnets exist (because the system-defined or clusterwide groups do not keep LIFs on their own subnets).
  - If you want to logically group a certain type of interface (for example, 10GbE-based LIFs fail over only to other 10GbE ports).
  - For LIFs that configured on top of VLAN ports and you want the LIFs to move to ports that can communicate with the other devices that are members of that same VLAN.
- **Clusterwide.** This type is automatically created during setup, and it cannot be modified—and it doesn't need to be. It includes all data ports in the cluster by default, and it is the default failover group for cluster management LIFs. As long as the network is flat (that is, there are no subnets), it successfully controls failover of the LIFs that are assigned to it.
- **System defined.** This type is also automatically created during setup, it cannot be modified, and it controls failover of all data LIFs by default. As with the clusterwide type of group, system defined is useful as long as the network is flat.

  Note: System-defined groups contain ports only from a maximum of two nodes: ports from one node of an HA pair combined with ports from a node of a different HA pair. This feature decreases the chance of complete loss of connectivity if some type of network issue causes one node in an HA pair to fail, followed by the second node in the same HA pair.

Figure 7) Default behavior for a system-defined failover group beginning with clustered Data ONTAP 8.2.



Best Practice: For Failover Groups in Clustered Data ONTAP 8.1.x to 8.2.x
NetApp recommends configuring user-defined failover groups and setting the failover policy to `nextavail`.

Failover Groups as of Clustered Data ONTAP 8.3.x

A failover group is automatically created when a broadcast domain is created. This automatically created failover group is associated with the broadcast domain and contains all the ports that belong to the broadcast domain. The name of this automatically created failover group is the same as for the broadcast domain. Therefore, when creating a new broadcast domain, clustered Data ONTAP takes care of creating the initial failover group. Any subsequent failover groups that are going to be associated with the same broadcast domain can be created and named appropriately (something standard to your environment).

In the case of user-created failover groups, the failover groups must be associated with a broadcast domain. All ports that are added to a failover group must be in a broadcast domain and must be in the same broadcast domain. Removing a port from a broadcast domain removes the port from all failover groups in the broadcast domain. If it's the last port in the failover group, the failover group is deleted. Deleting a broadcast domain deletes all failover groups in the broadcast domain.

In situations in which ports are removed from or added to a broadcast domain, the ports are automatically added to or removed from the autogenerated failover group. When the last port is removed from a broadcast domain, the automatically created failover group is deleted.

You must consider the failover policy that was chosen for the failover group, whether it's an automatically created failover group or a user-defined failover group. The failover policy determines how LIFs migrate or fail over if the need arises. Each policy type is discussed in the following sections; after you read through the policy types and then apply them to your environment, you can select a recommended best practice on a case-by-case basis.

Failover Policy Beginning with Clustered Data ONTAP 8.3

In Data ONTAP 8.3, the failover policy object has been extended to support new policy types. The failover policy dictates which among the targets within the failover group are selected as possible targets on a given LIF failover and the order in which the target list is traversed. The current set of failover policies has been extended to include:

- **local-only.** This type denotes that the targets should be restricted to the local or home node of the LIF. If you want to confirm that no I/O is accessed by using a remote path, NetApp recommends this type as a best practice.

- **sfo-partner-only.** This type denotes that the targets should be from the home node and its storage failover (SFO) partner only. If you are considering a performance-critical environment, NetApp recommends this type as a best practice.
- **broadcast-domain-wide.** This type denotes that all the ports that are owned by the same broadcast domain are candidates for failover. If maximum LIF availability is the most important consideration, NetApp recommends this type as a best practice.
- **system-defined.** This policy is the default for LIFs of type data. The policy includes targets from the home node and from every other node that the SVM spans, essentially skipping over the SFO partner. For example, in a six-node cluster, numbered 1 through 6, a LIF with a home node of 1 has the failover targets that include ports from nodes 1, 4, and 6. With a home node of 2, the ports selected are from nodes 2, 3, and 5. This approach allows rolling upgrades, rebooting either odd-numbered or even-numbered nodes at the same time. See Figure 7 for a graphic representation.
- **disabled.** This type denotes that failover has been disabled. Do not disable failover capabilities unless it is your intent to disable failover. Make sure that you don't miss this step during setup, upgrade, or creation, because it results in a support case that is simply a misconfiguration issue.

The following failover policy types have been deprecated:

- The `priority` and `nextavail` failover policies are no longer exposed through the UI. The functionality of `nextavail`, which is to stay local, is the default method of failing over the LIF. If for some reason the LIF can't fail over locally, the home node is the next preferred choice.

Table 1 lists the different types of LIFs in a clustered Data ONTAP system and the default failover policy associated with each.

Table 1) Default failover policies beginning with clustered Data ONTAP 8.3.

LIF Role	Default Failover Policy	Does It Need to Be Changed?
Cluster	`local-only`	No
intercluster	`local-only`	No
node-mgmt	`local-only`	No
cluster-mgmt	`broadcast-domain-wide`	No
Data	`system-defined`	It doesn't need to be changed but can be changed based on the needs of the environment

To check the failover policy setting, use the following command:

```
::> network interface show -fields failover-group,failover-policy
```

Or use the command below, which shows all LIF properties:

```
::> network interface show -vserver <vserver name> -lif <lif name>
```

To modify the failover policy setting, use the following command:

```
::> network interface modify
```

Failover Group: Examples for Verifying Configurations

Earlier than Clustered Data ONTAP 8.2

The following is an example of an incorrectly configured IFGRP, with explanations about why it is incorrectly configured and the steps to remedy it.


```
ontaptme-rtp::> network interface failover-groups create -failover-group scon_test1 -
node ontaptme-rtp-01 -port e0a
       (network interface failover-groups create)

ontaptme-rtp::> network interface failover-groups create -failover-group scon_test1 -
node ontaptme-rtp-01 -port e0b
       (network interface failover-groups create)

ontaptme-rtp::> network interface modify -vserver scon_test1 -lif scon_test_lif1
       -failover-group scon_test1
```

The failover group is not configured correctly because it is assigned to use the system-defined failover group (look at the `Use Failover Group` field).

```
ontaptme-rtp::> network interface show -vserver scon_test1 -lif scon_test_lif1
         (network interface show)

       Abbreviated output…

       Use Failover Group: system-defined

       Abbreviated output…

       Failover Group Name: scon_test1
```

You can execute the following steps to properly configure the failover group:

```
ontaptme-rtp::> network interface modify -vserver scon_test1 -lif scon_test_lif1 -use-
failover-group enabled -failover-group scon_test1
       (network interface modify)

ontaptme-rtp::> network interface show -vserver scon_test1 -lif scon_test_lif1
  (network interface show)

       Abbreviated output…

       Failover Policy: nextavail

       Abbreviated output…

       Use Failover Group: enabled

       Abbreviated output…

       Failover Group Name: scon_test1
```

To identify any NAS data LIFs that don't have failover groups configured, use the following command:

```
ontaptme-rtp::> network interface show -failover-group system-defined -data-protocol
nfs|cifs
```

Beginning with Clustered Data ONTAP 8.2

In clustered Data ONTAP 8.2, the "`Use Failover Group`" field is no longer visible in the "`network interface show`" output:

```
ontaptme-rtp::> network interface show –vserver test1 –lif testlif5
(network interface show)

       Abbreviated output…
```



```
       Failover Policy: nextavail

       Abbreviated output……

       Use Failover Group: This field does not exist in clustered Data ONTAP 8.2; see
following note.

       Abbreviated output…

       Failover Group Name: test-group
```

Note: Because the `Use Failover Group` field does not exist in clustered Data ONTAP 8.2, only the `Failover Policy` and `Failover Group Name` parameters must be configured correctly. Beginning with clustered Data ONTAP 8.2, the behavior that the `Use Failover Group` field used to control is now inferred by clustered Data ONTAP. That assumption can be made based on the values set for the other two relevant fields in clustered Data ONTAP 8.2.

Beginning with Clustered Data ONTAP 8.3

In clustered Data ONTAP 8.3, the "`Use Failover Group`" field has been removed from the output of the "`network interface show`" command entirely:

```
ontaptme-fc-cluster::> net int show -vserver mike_test -lif mike_test1
  (network interface show)

       Abbreviated output…

       Failover Policy: broadcast-domain-wide

       Abbreviated output…

       Failover Group Name: 10.228.225.37/24

       Abbreviated output…
```

Note: As with clustered Data ONTAP 8.2, the `Use Failover Group` field does not exist in clustered Data ONTAP 8.3; only the `Failover Policy` and `Failover Group Name` parameters must be configured correctly. Beginning with clustered Data ONTAP 8.2, the behavior that the `Use Failover Group` field used to control is now inferred by clustered Data ONTAP. That assumption can be made based on the values set for the other two relevant fields in clustered Data ONTAP 8.3.

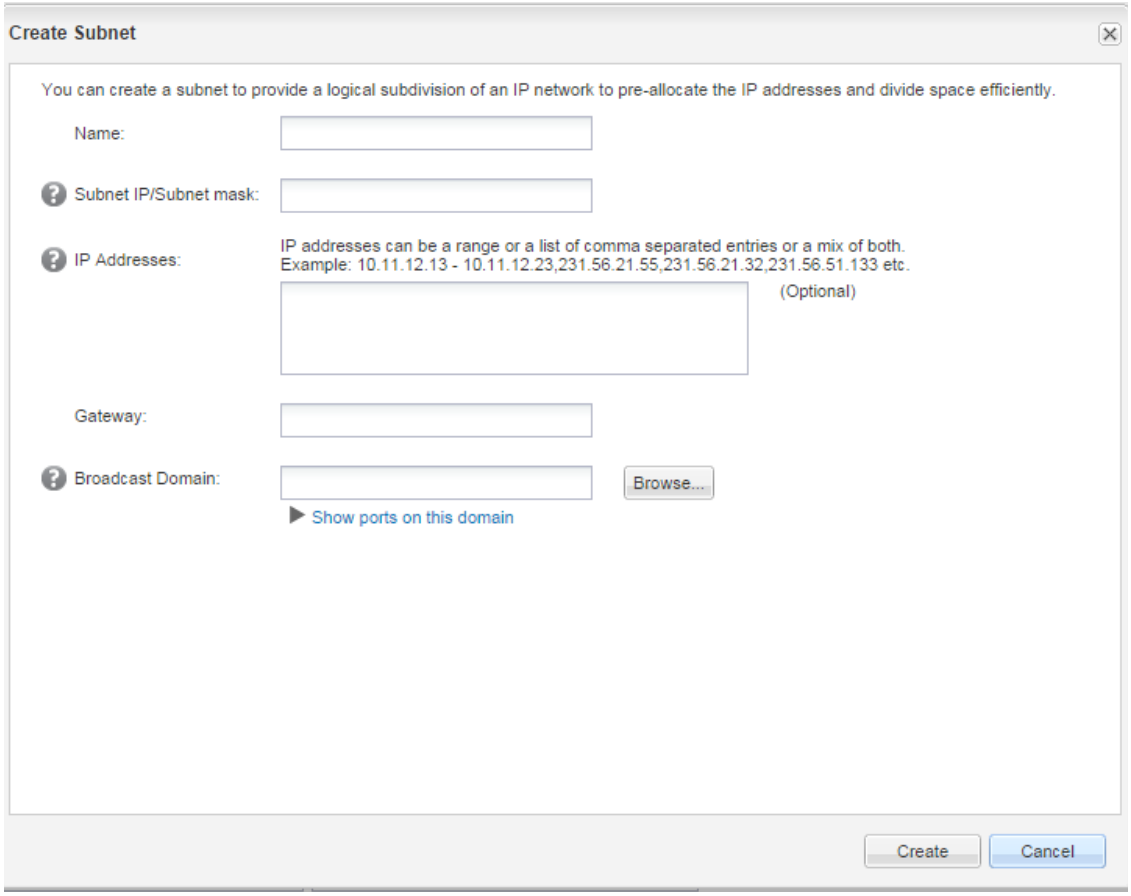
3.5 Subnet

The subnet (layer 3) is a named set of IP ranges with an associated gateway. Once referred to as an "IP pool," this object is owned by the broadcast domain and is optional. If you specify a subnet, you can preconfigure it with an IP address pool or you can choose to assign IPs manually as each LIF is created.

A subnet is composed of one primary component:

- **IP Address Range,** which is a pool of IP addresses that can be automatically provisioned when a new LIF is created. The addresses can be a range (contiguous), a comma-separated list, or a mix of both. See Figure 8 for an example of the Create Subnet window in NetApp OnCommand® System Manager.

Figure 8) Create Subnet window from OnCommand System Manager.



3.6 Storage Virtual Machine

In this section, we go into detail about the objects that are “owned” by an SVM:

- **LIFs.** Beginning in 8.3, node and cluster management LIFs can no longer be used by SVMs to connect to external resources. As our secure multi-tenancy networking design evolves, we are now able to provide the necessary connectivity without having to use a node or cluster management LIF.
- **Routing.** Beginning in 8.3, routing tables are defined per SVM.
- **Firewall policy.** Set it accordingly based on the type of access needed.
- **Failover group.** The SVM shares “ownership” of the failover group with the broadcast domain object.

Logical Interfaces

Logical interfaces (LIFs) are created as an abstraction on top of the physical (physical ports) or virtual interface (VLANs or IFGRPs) layer. IP-based LIFs for NAS or iSCSI (RFC 3720) are assigned IP addresses, and FC-based LIFs are assigned worldwide port names (WWPNs).

Data

The data LIF is used for data traffic (NFS, CIFS, FC, iSCSI). Although you use a data LIF for either NAS or SAN traffic, you cannot use the same data LIF for both. The NAS data LIF can fail over or migrate to other data ports throughout the cluster if configured to do so through a failover group. Also, as a very important distinction, NAS data LIFs migrate; SAN data LIFs (including iSCSI) do not migrate but instead use ALUA and MPIO processes on the initiators to handle path failures.

Note: Make certain that failover groups and the LIFs assigned to them are configured correctly. That is, you should configure the failover groups to use ports in the same subnet or VLAN and verify that LIFs are assigned to the correct failover groups. If ports from different subnets are used in the same failover group or if LIFs aren't assigned to the correct failover groups and a failover occurs, the result is loss of network connectivity. This loss of connectivity results in the loss of data availability.

Cluster

The cluster LIF can be configured only on 10GbE or 1GbE (1GbE can be used on the NetApp FAS2040 and FAS2220 platforms) ports of the type cluster. Also, it can fail over only to cluster ports on the same node.

The cluster LIF is used for operations such as:

- Moving volumes
- Synchronizing cluster or node configuration and metadata among the nodes in the cluster (this communication aspect is very important because it keeps nodes in the cluster in quorum)
- Accessing data on multiple nodes in the cluster
- Replicating intracluster data

For additional information, visit the Cluster Management and Interconnect Switches page.

Cluster Management

The cluster management LIF is used to manage the cluster. It can reside on and fail over to data ports only, but it can fail over to any data port on any of the nodes in the cluster. Therefore, make sure that this LIF is assigned to a correctly configured failover group.

Node Management

A node management LIF exists on every node in the cluster and is used for processes such as NetApp AutoSupport® diagnostics, SNMP, NTP, DNS, and other node-specific management traffic. (For a complete list of processes, see the Clustered Data ONTAP 8.2 Network Management Guide.) It can also be used to manage the node directly in the cluster for system maintenance purposes. It can fail over to other data or node management ports on the same node only. Therefore, make sure that this LIF is assigned to a correctly configured failover group.

Intercluster

The intercluster LIF is used for peering from cluster to cluster. These LIFs are node-specific; they can use or fail over to intercluster or data ports on the same node only. At least one intercluster LIF is required per node for replication between clusters. However, for the sake of redundancy, NetApp recommends as a best practice that you configure at least two intercluster LIFs or use an IFGRP. With an IFGRP, you do not have to dedicate two physical ports just for intercluster traffic. You should maintain consistent settings between the intercluster LIFs (same MTUs, flow control, TCP options, and so on).

The intercluster LIF must reside on a system SVM within the desired IPspace (it can be the default IPspace or a custom IPspace) and cannot be configured on data SVMs. This requirement is explained further in section 3.7, where cluster peering and IPspaces are described in detail. See section 9.1 for a sample output that shows the creation of an intercluster LIF.

SVM Management

An SVM management LIF is not a data-serving LIF, but, rather, it denotes a LIF that has the `role` set to `data` and the `data_protocol` set to `none`. This type of LIF is particularly important for clustered Data ONTAP 8.3 because of multi-tenancy; that is, all traffic is restricted to LIFs that belong to its SVM. Each

data SVM must have connectivity to services such as DNS, and so on, through LIFs that belong to that SVM. Beginning with clustered Data ONTAP 8.3, node management LIFs can no longer be used. Thus it is typical in an 8.3 deployment to configure several SVM management LIFs—one per data SVM—on the `e0M` management port, for example, so that data SVMs can access DNS and other services.

There is an important consideration to keep in mind when accessing external management services such as Active Directory or DNS. If the management and data ports are members of the same subnet and if firewall policies prevent **both** from communicating with the target, Data ONTAP might choose the incorrect LIF, resulting in network communication failures. NetApp recommends that management and data traffic be separated into different subnets, with routing policies defined within the SVM so that the correct LIF is used to communicate with these external services.

Best Practice: SVM Data and Management Traffic
Segregate management and data traffic into different subnets and configure routing policies appropriately so that the correct SVM LIFs are used to communicate with external services.

Table 2 lists the different LIF types in a clustered Data ONTAP system, their associated function, and limits associated with each.

Table 2) Additional information regarding LIFs.

LIF Type	Function	Minimum Required	Maximum Allowed
Node management	Used for system maintenance of a specific node, SNMP, NTP, and NetApp ASUP™ tool	1 per node	1 per port/subnet
Cluster management	Management interface for the entire cluster	1 per cluster	N/A
Cluster	Used for intracluster traffic	2 per node	2–4 per node depending on the platform; check the Hardware Universe for specifications
Data	Associated with an SVM and used for data protocols and protocol services (NIS, LDAP, Active Directory, Windows Internet Name Service, DNS)	1 per SVM	128 per node in HA configuration 256 per node in non-HA
Intercluster	Used for intercluster communication, such as setting up cluster peers and NetApp SnapMirror® traffic	1 per node if cluster peering is enabled	N/A
SVM management	Not a data-serving LIF, but can be used for external resource connectivity (Active Directory, DNS, and so on) for the SVM	1 per SVM	Restricted by the 256 LIF limit per node in non-HA/128 per node in HA

Routing Changes in Clustered Data ONTAP 8.3

As explained earlier in this document, routing changes were made between clustered Data ONTAP 8.2 and 8.3. Beginning in clustered Data ONTAP 8.3, routing is now specific to an SVM. Each SVM owns a

set of routing tables. This model is slightly different from the one in clustered Data ONTAP 8.2, in which the routing tables were referenced as a route per SVM per subnet per role. Routes are not shared between SVMs. One exception is IPv6 dynamic routes. In the event of IPv6 traffic that uses dynamic routing, those requests still use a model that is similar to clustered Data ONTAP 8.2.

Other things to take into consideration when configuring SVM LIFs include:

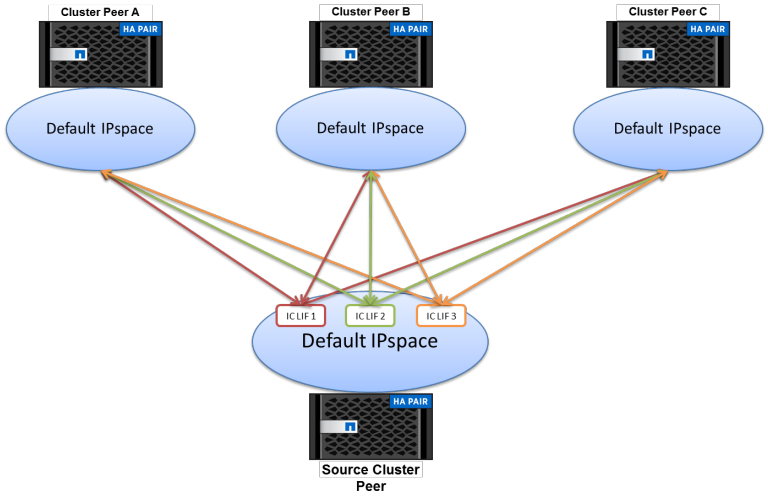
- **Firewall policy.** Make certain that the correct policy is set based on the functionality of the LIF; for more information, see the network management guide.
- **Failover group.** As previously mentioned, the broadcast domain object "owns" the failover group object. However, the SVM also loosely "owns" the failover group object. The result is that within an IPspace, the SVM and the broadcast domain objects both share ownership overall for a failover group. When LIFs are created and are assigned to their respective SVMs, a failover group is assigned to each LIF accordingly.

3.7 Cluster Peering and IPspaces

In versions earlier than clustered Data ONTAP 8.3.1, cluster peering was required to go through the default IPspace. In addition, all intercluster LIFs had to reside in the default IPspace and were required to have full-mesh connectivity with peer-cluster intercluster LIFs. For multi-tenant environments and service providers, these requirements can present a challenge if tenants reside on disparate networks because the full-mesh requirement is violated, and the peering relationship is considered unhealthy.

See Figure 9 for the cluster peering structure in versions earlier than clustered Data ONTAP 8.3.1.

Figure 9) Multiple cluster peers in versions earlier than clustered Data ONTAP 8.3.1.



In clustered Data ONTAP 8.3.1, these constraints have been lifted with the ability to establish cluster peering relationships in custom-defined IPspaces. Intercluster LIFs can be created in the desired IPspace, and the full-mesh connectivity requirement exists only within the IPspace of the peering relationship. Additionally, the data LIFs of the SVM do not have to reside on the same IPspace as the intercluster LIFs that are used for peering. This freedom presents significant advantages for customers who require more advanced control of the clustered Data ONTAP storage cluster in multinetwork and multi-tenant environments. It also simplifies how cluster peering relationships and intercluster LIFs are established when multiple IPspaces exist on a cluster.

When configuring cluster peering relationships with custom-defined IPspaces, it's important to understand the different types of SVMs and their respective roles:

- Data ONTAP 8.3 introduced a new type of SVM: a *system* SVM.
- Although a system SVM might look like a data SVM, its function is actually very different:

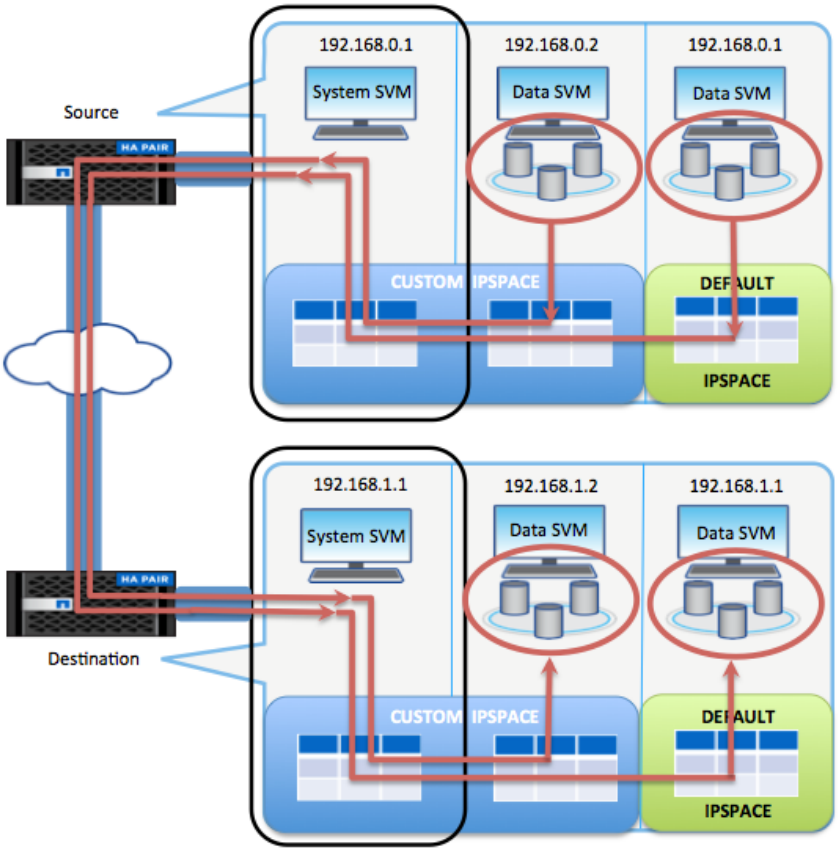
  - A system SVM is a container for network configuration information for cluster communications.
  - System SVMs can contain intercluster LIFs; data SVMs cannot.
  - System SVMs do not have associated volumes.
  - System SVMs **cannot** host data LIFs.
- All the networking of an SVM is scoped to only one IPspace. This approach is true for both system SVMs and data SVMs.

With this information in mind, remember that when you create cluster peering relationships by using either default or custom IPspaces, the intercluster LIFs must reside within the system SVM, not the data SVM.

In the example shown in Figure 10, the administrator has decided to create a cluster peering relationship between two custom IPspaces. Data SVMs in the custom and the default IPspaces consult their respective routing tables and forward replication traffic to the system SVM in the custom IPspace. The system SVM in turn forwards the traffic off to the peered cluster.

Note: In Data ONTAP 8.3.1, only a single peering relationship can exist between two clusters.

Figure 10) Cluster peering with custom IPspaces in clustered Data ONTAP 8.3.1.



3.8 LIF Sharing for Outbound Connections

Beginning with clustered Data ONTAP 8.3, there is a better definition of uniform network access between nodes in a cluster. There might be cases in which nodes in a cluster have an SVM spread out among all the nodes, but not all nodes have the data LIFs necessary for the SVM to pass traffic out of the cluster.

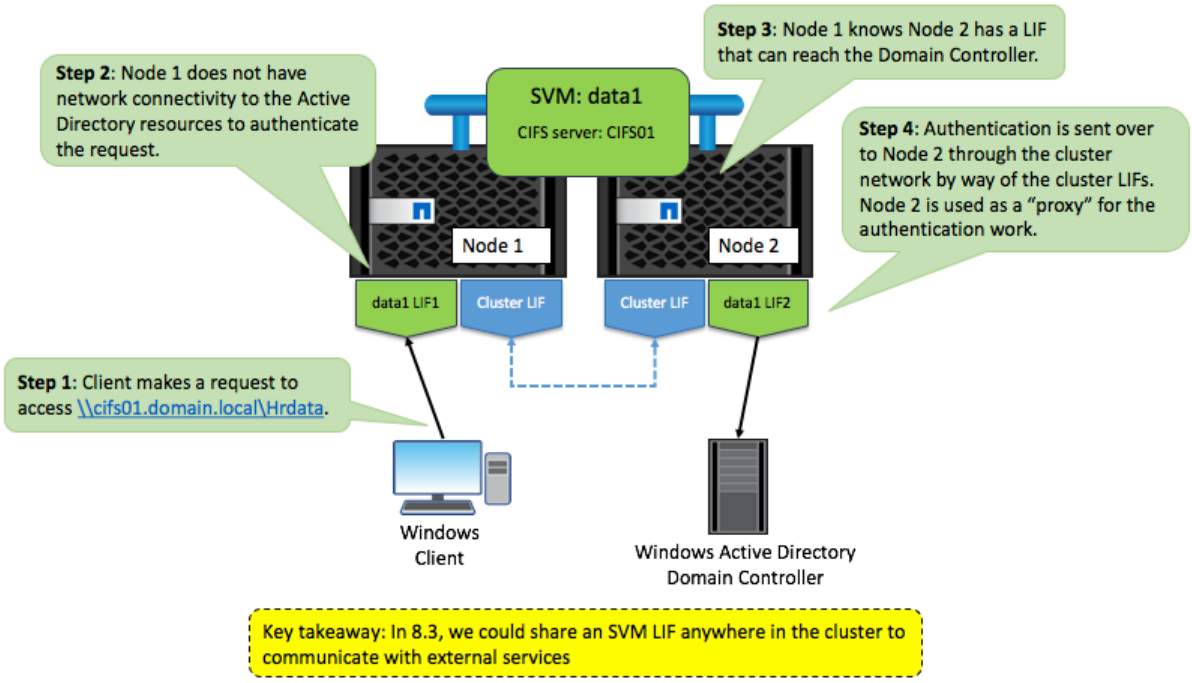
For example, it is possible that an SVM has some of its LIFs connected to a dedicated management network. In that case, certain network services (for example, DNS, LDAP, NIS, and so on) might be

reachable only on the management network. If an SVM has a single LIF on the management network, the management services are reachable from only a single node.

These differences in connectivity make it impossible to assume that code running on a given node is able to connect to network services on behalf of a particular SVM. The new networking functionality in Data ONTAP 8.3 closes this gap. It transparently tunnels packets through the cluster network between the nodes without a data LIF to one of the other nodes in the cluster that does have a data LIF. The tunneling occurs over the cluster network by using the cluster LIFs.

Figure 11 shows a scenario in which Active Directory services are available from only one clustered Data ONTAP node. It also shows the steps that Data ONTAP uses to communicate with the service from another node in the cluster.

Figure 11) Continuing evolution of secure multi-tenancy networking.



For the majority of use cases, this new LIF routing functionality eliminates the need to have a data LIF configured per SVM per node (as has been suggested in past clustered Data ONTAP releases). There are exceptions in which this new functionality does not work and a "data LIF/SVM/node" model might still be needed. The use cases for both are listed in Table 3.

Table 3) Clustered Data ONTAP 8.3 secure multi-tenancy networking use cases described.

Use Case	How Request Is Processed
• User/group quotas • Ping and traceroute • Name service: LDAP, NIS, Active Directory, iSNS, and so on • Management plane triggers such as CIFS server creation, CIFS share ACL modification, host name lookups from CLI • Applying CIFS Group Policy Objects	These types of requests are managed by the new model.

Use Case	How Request Is Processed
• NetApp FPolicy® component	The FPolicy application intentionally disables LSOC for its sockets, and therefore use of this feature requires a LIF per SVM per node.
• Cluster peering	In the case of cluster peering, all traffic from each SVM uses the default IPspace to transfer to the destination cluster. In this case, the new model is not used because each SVM is required to have an intercluster LIF configured.
• NDMP	In the case of NDMP, traffic from each SVM (that is using NDMP for backup) uses the default IPspace to transfer to the destination device. In this case, the new model is not used because each SVM is required to have the appropriate type of LIF configured.

4 Network Design Considerations

4.1 DNS Load Balancing

DNS load balancing in its different forms takes advantage of different types of algorithms to determine the optimal LIFs to return resolution requests. Two different types of DNS load balancing are currently supported with NetApp clustered Data ONTAP: zoning based and round-robin. By using zoning based (commonly referred to as *on box*), you can create a DNS load-balancing zone on the SVM that returns a lower-loaded LIF. Which LIF is returned is based on the network traffic and the availability of the port resources (CPU usage, throughput, open connections, and so on). With round-robin-based DNS load balancing (commonly referred to as *off box*), you use a round-robin-based algorithm to determine which resources respond to resolution requests.

Also, for either option, only new NAS mounts are affected. Any existing mounts or shares must be remounted to take advantage of the load-balancing benefits. For details about setting up either of these options in an NFS environment by using different types of authentication, see TR-4067: Secure Unified Authentication with NetApp Storage Systems, beginning on page 23.

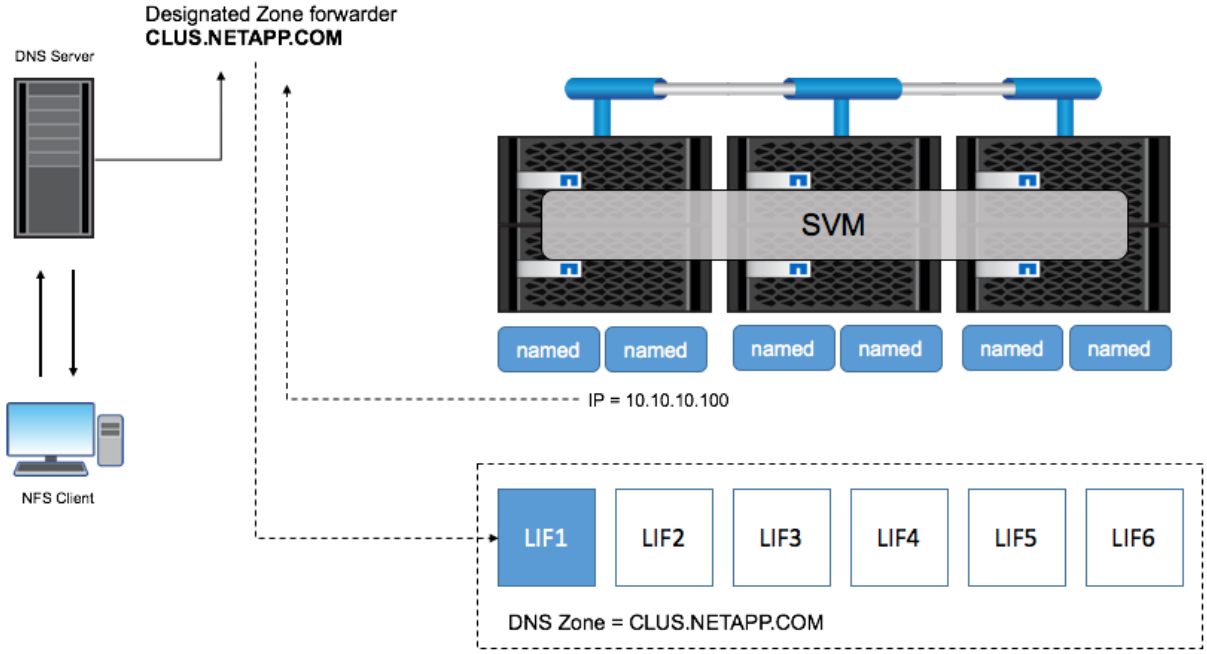
Zoning Based/On Box

In this configuration, there might be a measurable increase in performance with certain workloads. However, it is a bit more complex to set up and manage. After the delegation zone is added to the sitewide DNS server and the data LIF IP addresses are added to it, however, the storage administrator can manage it from then on. This capability could decrease dependency on the network team. (Note: You are adding data LIF IPs that belong to each individual SVM, not cluster management, cluster, or node management.) The functionality is protocol-agnostic and works with the different versions of NFS and SMB.

With this configuration, you delegate queries to the DNS server (`named`) that runs on each node inside the cluster, which then passes it to the individual SVMs. At that point, clustered Data ONTAP calculates the load of each data LIF automatically and then passes the query to the appropriately balanced LIF.

In the example in Figure 12, the NFS client queries the sitewide DNS server for resolution of the share that it wants to mount from the SVM (`SVM`). The sitewide DNS server passes the query by using the delegation zone `CLUS.NETAPP.COM` to the `named` process running on the cluster, which resolves access based on the appropriately balanced data LIF in the SVM. Keep in mind that if an environment has many SVMs, you must account for each of the data LIFs of each of the SVMs that are added to the delegation zone on the sitewide DNS server.

Figure 12) DNS load balancing: zoning based.

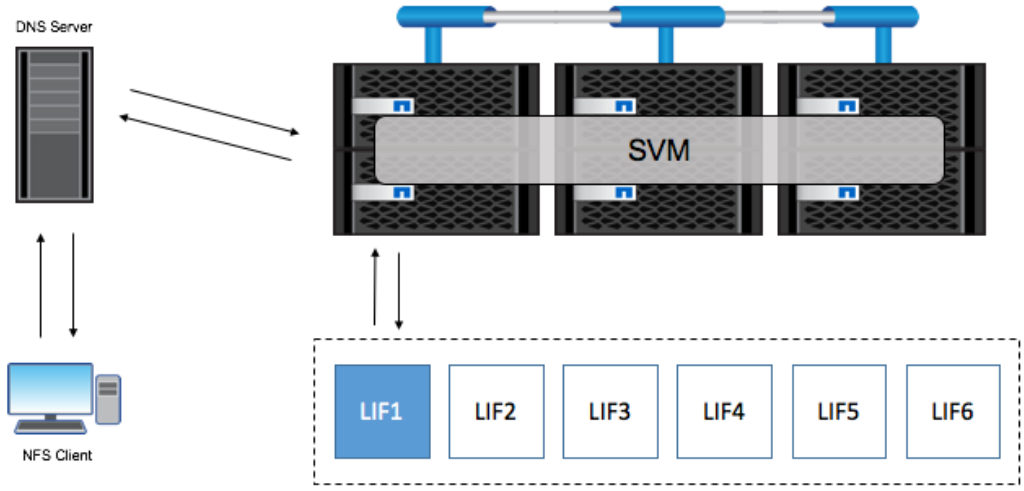


Round-Robin/Off Box

In this configuration, the setup and management are a bit easier compared with the zoning-based/on-box configuration. However, issues such as client and DNS hierarchy caching could cause resource bottlenecks. Also, the round-robin algorithm might not be as effective as the algorithm used by clustered Data ONTAP. It might be a good solution, however, if the environment is uniformly configured (similar types of servers, evenly distributed connections). It might also be a better solution if, for example, the same team manages the DNS and storage environments.

In the example in Figure 13, each data LIF in each SVM that resides in the cluster has a DNS “A” record that was created with the same name. The NFS client makes a request for name resolution to the sitewide DNS server. The sitewide DNS server resolves the request to an IP address by using a round-robin algorithm. The sitewide DNS server responds to the client with the chosen IP address. Keep in mind that if an environment has many SVMs, you must account for each of the data LIFs of each of the SVMs that are added to the sitewide DNS server.

Figure 13) DNS load balancing: round-robin.



4.2 Automatic LIF Rebalancing

Automatic LIF rebalancing can be used to allow clustered Data ONTAP to dynamically migrate LIFs that reside on overused ports to ports with lower use. Automatic LIF rebalancing calculates a weighted, balanced distribution of load across the ports. This value is then automatically assigned to the LIFs based on statistics about the current node and port resources:

- **Follows the failover group rules.** Make certain that the failover groups are configured correctly so that failovers occur as expected. Specifically, as a required best practice, be sure that the ports that make up the failover groups are part of the same subnet. That way, the LIFs do not lose connectivity if they are moved around or are rebalanced within the failover group.
- **Only works with NFSv3.** If the connection is moved, the I/O request resumes after the connection is reestablished.
- **Other traffic.** If any other NAS traffic exists with NFSv3 on a LIF, it negates autorebalancing.

4.3 IPv6 and Clustered Data ONTAP

IP version 4/RFC 791 (IPv4) was the first widely used standard to define how the Internet and its connected devices operated and communicated with one another. When IPv4 became the Internet standard, the 4.2 billion possible IP addresses were never intended to house a global commercial Internet. It was 1981, and only a limited number of computers needed to connect to the Internet (mostly U.S. government and research entities); web-capable phones, tablets, glasses, and so on, were far from being invented. This pool of IP addresses has been in use for the entire history of the commercial Internet, but constantly evolving technology has driven the available IP address pool very close to depletion.

The solution to IPv4 address depletion is IP version 6/RFC 2460 (IPv6). IPv6 holds 340,282,366,920,938,463,463,374,607,431,768,211,456 IP addresses. This exponentially larger pool of IP addresses is the key to the future growth of the global network infrastructure, and companies that use and distribute IP addresses must adapt their networks and systems to use IPv6.

NetApp has supported protocols that use IPv6 to communicate to FAS controllers in the clustered Data ONTAP codeline since clustered Data ONTAP 8.2.x. NetApp has always balanced the needs of its customers with the right amount of effort and support in regard to IPv6. As technology advancements and

customers' demands have increased, NetApp has dedicated more and more effort into the development and support of IPv6 in the clustered Data ONTAP codeline.

As adoption rates increase globally, NetApp will continue to support the innovation of the protocol and the adoption of it within the solutions that NetApp provides to its customer base. The following features and considerations should be referenced for any technical-type discussions about NetApp and IPv6:

- Features:
  - Resolution of any issues regarding IPv4 address depletion.
  - Autconfiguration capabilities: Beginning with clustered Data ONTAP 8.2, all autoconfiguration options are supported except for stateless address autoconfiguration (SLAAC). Support for SLAAC will be added in a future release, with the ability to enable or disable it as needed; however, a time frame hasn't been committed yet.
  - End-to-end connection integrity.
  - Multicasting ability.
  - More efficient routing.
  - Simplified packet headers that make packet processing more efficient.
- Considerations:
  - Network address translation (NAT) is not a security feature; it breaks the end-to-end communication model and adds an unnecessary layer of frustration to an administrator's work. NetApp recommends that the use of NAT be considered an exception; if IPv6 is going to be implemented into an environment, it should be all IPv6.
  - Dual-stack support with cluster peering: When creating peering relationships between two clusters, either IPv4 or IPv6 LIFs must be used; it cannot be a mix of each.
  - Dual-stack support in general: If an SVM is connected to the same network, all of its LIFs can use only IPv4 or IPv6, but not both. If the SVM is connecting to two different networks, it can have separate LIFs configured: one set of LIFs to connect to the IPv4 network and another set to connect to the IPv6 network.
  - There could be security concerns if a hacker is able to access and obtain an IPv6 address on your network. Because the IPv6 range is so large, a hacker could continuously allocate a new IPv6 address, making it more difficult to locate and remove the hacker from your environment.
- Transition topics: If measured by industry network standards, a pure IPv6 network is preferred if adoption is acted upon. However, NetApp understands that there can be challenges when making major changes and that there are cases in which a dual-stack configuration is necessary.

Regardless of whether your organization chooses pure IPv6 adoption initially or a dual-stack configuration, a solid transition plan must be developed to help with the transition. Following are some topics to consider:

- **Outside in.** Start by providing external users with IPv6 public access to your services—for example, by running a reverse proxy that handles IPv6 customers—and then progressively enable IPv6 internally.
- **Inside out.** Start by enabling internal networking infrastructure, hosts, and applications to support IPv6. Then progressively reveal IPv6 access to external customers.

4.4 Directory/Name Services

Directory services were part of an Open Systems Interconnection initiative to get everyone in the industry to agree to common network standards to provide multivendor interoperability. In the 1980s, the ITU and ISO came up with a set of standards, X.500 (the directory service standard). X.500 was intended initially to support the requirements of intercarrier electronic messaging and network name lookup.

Over the years, different forms of directory service implementations have been produced by different vendors. A directory service, called a *naming service*, maps the names of network resources to their respective network addresses. In this section, we discuss the more common options used:

- Domain Name System (DNS): RFC 1035

The DNS is a hierarchical, distributed database that contains mappings of DNS domain names to various types of data, such as IP addresses. DNS allows you to use friendly names, such as www.netapp.com, to easily locate computers and other resources on a TCP/IP-based network.

- Dynamic Domain Name System (DDNS or DynDNS): RFC 2136

DDNS is a service that is responsible for automatically updating a name server in the DNS with the active DNS configuration of its configured host names, addresses, or other information. It allows the name server to point to the same network host in the event of an IP address change.

Note: Data ONTAP 8.3.1 introduced support for DDNS in clustered Data ONTAP configurations. Customers can use DDNS to dynamically update their name servers when network configuration information changes on both SVMs and LIFs. With this capability, the master DNS has current and accurate DNS information.

- Network Information Service (NIS): RFC 3898

NIS is a distributed database that allows you to maintain consistent configuration files throughout your network.

NIS was developed independently of DNS and has a slightly different focus. Whereas DNS focuses on making communication simpler by using machine names instead of numerical IP addresses, NIS focuses on making network administration more manageable by providing centralized control over a variety of network information. NIS stores information not only about machine names and addresses, but also about users, the network itself, and network services. This collection of network information is referred to as the *NIS namespace*.

- Lightweight Directory Access Protocol (LDAP): RFC 4511

LDAP is a protocol for querying and modifying X.500-based directory services that run over TCP/IP. It is a lightweight alternative to the X.500 Directory Access Protocol (DAP) for use on the Internet. Several notable implementations have been built on the foundation of LDAP, including:

  – Active Directory (Active Directory is based on the LDAP/X.500 implementation).

  – Active Directory Domain Services (ADDS) provides a distributed database that stores and manages information about network resources and application-specific data from directory-enabled applications. Administrators can use ADDS to organize elements of a network, such as users, computers, storage equipment, and other devices, into a hierarchical containment structure. It acts to provide LDAP-based authentication with Kerberos (RFC 4120)-based authorization.

  – OpenLDAP is a free, open-source implementation.

- Internet Storage Name Service (iSNS): RFC 4171

The iSNS protocol is used for interaction between iSNS servers and iSNS clients. iSNS clients are computers, also known as *initiators*, that are attempting to discover storage devices, also known as *targets*, on an Ethernet network. iSNS facilitates automated discovery, management, and configuration of iSCSI and FC devices (by using iFCP gateways) on a TCP/IP network.

Best Practice: Name Services

NetApp recommends that you use the appropriate name services configuration for a particular environment, regardless of its complexity. For example, in a smaller and less complex environment, a name services setup allows the environment to scale as needed without keeping track of host and configuration files from several different sources throughout the environment.

For information about configuring name services in a clustered Data ONTAP environment, see the following documents:

- TR-4067: Clustered Data ONTAP NFS Implementation Guide
- TR-4191: Best Practices Guide for Clustered Data ONTAP 8.2.x and 8.3.x Windows File Services

5 Interoperability

The NetApp Interoperability Matrix Tool (IMT) enables you to search for information about the configurations for NetApp products that work with third-party products and components that meet NetApp specified standards and requirements. You can access the IMT at http://mysupport.netapp.com/matrix/.

6 Performance

6.1 Ethernet Flow Control

Ethernet flow control is a layer 2 network mechanism that is used to manage the rate of data transmission between two endpoints. It provides a mechanism for one network node to control the transmission speed of another so that the receiving node is not overwhelmed with data.

To accomplish this task, the IEEE 802.3x standard included a PAUSE frame, which when sent by a destination endpoint, would request that the sending node pause transmission for a specific time period. Although PAUSE frames were initially intended for NICs that had insufficient buffering to handle receiving full-speed network flows, one significant drawback was found during network congestion events within a switch. Because multiple network flows can be sent over a single switch port, a PAUSE frame would cause all flows over a link to be delayed, even those flows that were not causing congestion.

A subsequent effort by Cisco, *priority flow control* (defined in IEEE 802.1Qbb), was intended to replace 802.3x PAUSE and provide a mechanism that could be controlled independently for each class of service defined by IEEE 802.1p. Instead of pausing all traffic on a network link, priority flow control allows the administrator to selectively pause traffic according to its class of service.

The limitations of buffer designs on industry switches and of a shallow receive buffer on the Chelsio NIC, which shipped with many of our platforms, influenced NetApp recommendations for flow control. Historically, NetApp had recommended that flow control be disabled on all network ports within a NetApp Data ONTAP cluster. This approach is no longer the case. Guidance in this area has since changed, and the new recommended best practice is as follows:

- Disable flow control on cluster network ports in the Data ONTAP cluster.
- Flow control on the remaining network ports (the ports that provide data, management, and intercluster connectivity) should be configured to match the settings within the rest of your environment.

Best Practice: Ethernet Flow Control

NetApp recommends disabling flow control on all cluster network ports within a Data ONTAP cluster. NetApp makes no other recommendations for best practices for the remaining network ports. You should enable or disable as necessary.

Following are some sample outputs to help give perspective between the different scenarios that might come up in a flow control–related conversation.

The following is a sample output for block-related options. For example, a customer is asking about FCoE/priority flow control:

```
ontaptme-rtp::> fcp adapter modify ?
    -node <nodename>               Node
   [-adapter] <text>               Adapter
  [ -speed {1|2|4|8|10|16|auto} ]  Configured Speed
```



```
  [ -state {down|up} ]              Configured State
```

Note: There are no options to configure any type of flow control. Priority flow control occurs at the switch layer with a switch that is Data Center Bridging capable.

The following is an example output from ports that are available for nonblock traffic. For example, a customer is asking about the recommendation to disable flow control on cluster network ports:

```
ontaptme-rtp::> net port modify ?
  (network port modify)
   [-node] <nodename>                                        Node
   [-port] {<netport>|<ifgrp>}                               Port
  [[-role] {cluster|data|node-mgmt|intercluster|cluster-mgmt}]  Role
  [ -mtu <integer> ]                                         MTU
  [ -autonegotiate-admin {true|false} ]                      Auto-Negotiation Administrative
  [ -duplex-admin {auto|half|full} ]                         Duplex Mode Administrative
  [ -speed-admin {auto|10|100|1000|10000} ]                  Speed Administrative
  [ -flowcontrol-admin {none|receive|send|full} ]            Flow Control Administrative
← Here you see we have the option available to change the 802.3x flow control setting.
```

Note: Currently, by default in clustered Data ONTAP, when creating or configuring an interface on a NetApp controller, the default for flow control settings is `on` for both `send` and `receive`. Port settings are changed by using the `network port modify` command.

6.2 Jumbo Frames

Jumbo frames are Ethernet frames with a greater than 1,500-byte maximum transmission unit (MTU) (RFC 894) payload. In a clustered Data ONTAP environment, ports with a role type of cluster must be set to an MTU size of 9,000. If the cluster ports are set to a 1,500 MTU size, the cluster might operate but does so at a suboptimal level. Also, keep in mind that if the MTU is changed while the cluster port is active, the NIC resets and connections are dropped, which could have a very detrimental effect on the cluster.

Note: When a port is configured as the type cluster, during initial setup, the clustered Data ONTAP setup wizard sets it to 9,000 MTUs.

Note: All ports associated with a broadcast domain must have the same MTU size. When the MTU in a broadcast domain is modified, all ports associated with the broadcast domain are modified to the same value. If an attempt is made to modify the MTU on a port by using `net port modify`, which is associated with a broadcast domain, an error is usually returned. An error is not returned only if the port is being modified to the MTU of the broadcast domain. This feature helps MTU settings remain consistent in the event of some type of failure that causes the cluster to lose track of nodes, for example, if a node drops out of quorum.

6.3 Topology

Following are some guidelines for topology:

- To avoid undesirable performance degradation during a physical port failover, do not configure IFGRPs with different link speeds. For example, avoid creating an IFGRP and adding a mix of 10GbE and 1GbE interfaces to it. Instead, create the IFGRPs with like interfaces (for additional information, see Figure 14).
- When you add interfaces to an existing IFGRP, review the entire configuration in the environment from end to end. For example, let's say that you add two 10GbE interfaces to an existing IFGRP that contains two 10GbE interfaces (four interfaces total, or 40Gb aggregated). You should review the port group configuration on the upstream switches and on the network infrastructure. Your review should confirm that there is adequate bandwidth to prevent a possible bottleneck at the port group level on the switches (see Figure 15).
- You should verify that all interfaces and network ports that participate in the environment are correctly negotiating (auto, half, full, and so on) as expected.

- To check the settings for the ports for each node, use the following command on the nodes in the cluster:

```
network port show
```

- For the switches in the environment, use commands such as the following to check relevant port settings on the switches. (The following is an example only; see the documentation for your specific switch for the exact syntax):

```
show interface ethernet 1/1
```

- Do not use interface `e0M` (or any other port that is designated for node management) for any type of protocol traffic other than node management. This interface should be exclusively for node management and administrative access.

Figure 14) Do not mix port speeds in IFGRPs.

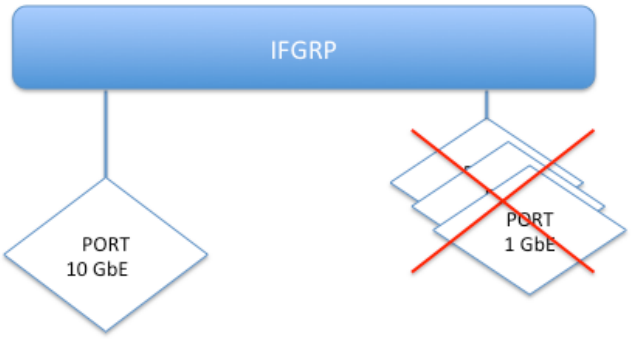
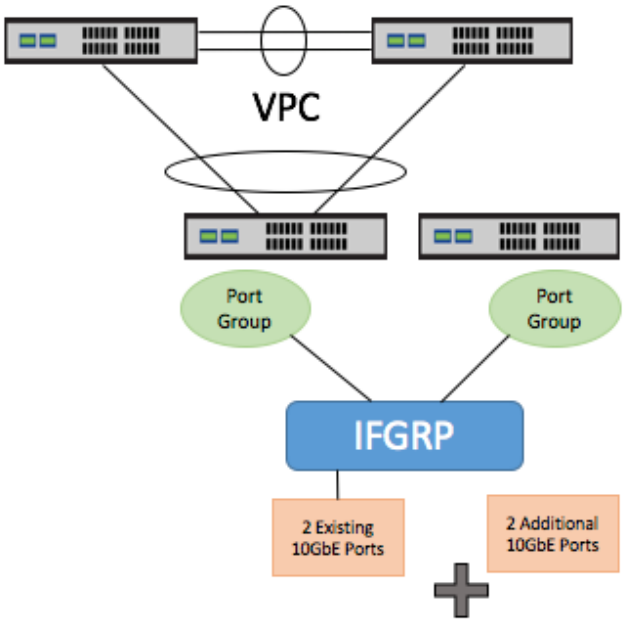


Figure 15) Review topology end to end if you add resources at any one point.



6.4 Volume and LIF Placement

You should take into account performance considerations when you determine volume and LIF placement within your cluster. When data LIFs that reside on a cluster node are used to access volumes

within aggregates that are hosted elsewhere, I/O requests are serviced by sending traffic across the cluster network. This approach might be acceptable under certain scenarios, such as for an underused cluster. However, at larger scale and as the cluster is put under increasing load, the presence of both direct and indirect I/O on your cluster can cause unanticipated performance bottlenecks. To prevent client throughput from being reduced by network saturation caused by increased cluster network traffic, it's important to monitor the amount of cluster network traffic that is generated under various workload scenarios.

Best Practice: Volume and LIF Placement

When determining volume and data LIF placement, to reduce the amount of cluster network traffic generated, you should confirm that high traffic volumes and LIFs are placed on the same cluster node. Cluster network saturation can result in severe performance degradation with sequential read workloads when data volumes are remotely accessed (when the volume and client accessed data LIF are not co-resident).

It should be noted that certain operations, such as volume move, will generate a significant amount of cluster network traffic, further compounding this scenario. When investigating whether volume and LIF placement might be contributing to cluster network saturation, it's best to confirm that there are no active volume move operations. Furthermore, confirming that sufficient cluster network bandwidth is available to service anticipated volume move traffic is important and can prevent future performance bottlenecks or even failed move operations.

As cluster network traffic congestion increases, to improve load-balancing and reduce the amount of traffic generated by a single interface, it might be beneficial to increase the number of cluster network ports. Monitoring the network use of individual cluster network interfaces while the system is under load will help when making this determination. You can perform this monitoring through the use of several performance counters that are available within Data ONTAP, specifically:

- **Object:**
  - `nic_common`
- **Counters:**
  - `rx_bytes_rate`
  - `tx_bytes_rate`

Best Practice: Performance Counter Analysis and Remediation

Analyze the results of the preceding performance counters. Confirm that there are no active volume move operations. If the network traffic crossing the cluster network ports is greater than that of the data LIF ports, it's likely that the volume and LIF placement of the system is suboptimal. In that case, the data LIF should be migrated to the cluster node that owns the volume's hosting aggregate.

The following example leverages these counters to illustrate the troubleshooting methodology that is used to identify whether cluster network congestion is a potential issue. In this scenario, the SVM data LIF and volume are owned by different cluster nodes.

```
ontaptme-rtp::*> vol show -vserver lippe -volume lippe_nfs -fields aggregate
vserver volume    aggregate
------- --------- ---------
lippe   lippe_nfs aggr1

ontaptme-rtp::*> aggr show -aggregate aggr1 -fields owner-name
aggregate owner-name
--------- ---------------
aggr1     ontaptme-rtp-01

ontaptme-rtp::*> network interface show -vserver lippe
            Logical    Status     Network            Current       Current Is
Vserver     Interface  Admin/Oper Address/Mask       Node          Port    Home
----------- ---------- ---------- ------------------ ------------- ------- ----
lippe
```



```
            lippe_data   up/up    10.228.225.130/24  ontaptme-rtp-02
                                                                  e0a     true

ontaptme-rtp::*> network interface show -role cluster -home-node ontaptme-rtp-01
            Logical    Status     Network            Current       Current Is
Vserver     Interface  Admin/Oper Address/Mask       Node          Port    Home
----------- ---------- ---------- ------------------ ------------- ------- ----
Cluster
            ontaptme-rtp-01_clus1
                         up/up    169.254.108.205/16 ontaptme-rtp-01
                                                                  e1a     true
            ontaptme-rtp-01_clus2
                         up/up    169.254.227.148/16 ontaptme-rtp-01
                                                                  e2a     true
2 entries were displayed.

ontaptme-rtp::*> statistics start -object nic_common

Warning: The existing sample will be overwritten for sample-id: :sample_50094.
Do you want to continue? {y|n}: y
Statistics sample deleted, for sample-id: sample_50094
Statistics collection is being started for sample-id: sample_50094

ontaptme-rtp::*> statistics show -object nic_common -instance e0a -counter rx_bytes_rate -node
ontaptme-rtp-02

Object: nic_common
Instance: e0a
Start-time: 7/7/2015 13:23:09
End-time: 7/7/2015 13:23:51
Elapsed-time: 42s
Node: ontaptme-rtp-02

    Counter                                                     Value
    -------------------------------- --------------------------------
    rx_bytes_rate                                             8.70MB

ontaptme-rtp::*> statistics show -object nic_common -instance e2a -counter rx_bytes_rate -node
ontaptme-rtp-01

Object: nic_common
Instance: e2a
Start-time: 7/7/2015 13:29:21
End-time: 7/7/2015 13:30:45
Elapsed-time: 84s
Node: ontaptme-rtp-01

    Counter                                                     Value
    -------------------------------- --------------------------------
    rx_bytes_rate                                             4.33MB

ontaptme-rtp::*> statistics show -object nic_common -instance e1a -counter rx_bytes_rate -node
ontaptme-rtp-01

Object: nic_common
Instance: e1a
Start-time: 7/7/2015 13:29:21
End-time: 7/7/2015 13:30:58
Elapsed-time: 97s
Node: ontaptme-rtp-01

    Counter                                                     Value
    -------------------------------- --------------------------------
    rx_bytes_rate                                             4.30MB

ontaptme-rtp::*>
```


7 Upgrading from Clustered Data ONTAP 8.2 to 8.3

To upgrade to Data ONTAP 8.3, first determine whether sufficient LIFs and routes exist for each SVM, independent of the home nodes of the LIFs:

1. Run `system image update` to validate that each SVM has sufficient LIFs to reach all external servers.

Note: If there are transient conditions in the cluster while the script is being run (configurations changing, network connectivity problems occurring, and so on), the script might return unexpected results. Rerunning the script after these conditions are resolved should yield correct results.

2. Add LIFs and routes to each SVM until the number is sufficient.
3. Perform some advanced planning at this stage to save a little effort later.

   For example, decide on the first two nodes to be upgraded:

- Select ports on these two nodes that the LIFs mentioned in step 2 could use. Create a broadcast domain containing these ports; failover groups are created automatically containing the ports in the broadcast domain.
- When the LIFs in step 2 are modified or created, use the new failover groups that you have created.

Note: For further detailed information, see the upgrade guide.

8 Troubleshooting

8.1 Path MTU Black Hole Detection

What Is Path MTU?

As an IP packet travels across a network on its way toward its destination, it traverses a series of switches and routers, each with its own port MTU settings. As explained earlier in this document, the port MTU setting is the maximum transmission unit, or the largest IP packet, that is allowed by that port. Path MTU (PMTU), on the other hand, is the maximum transmission unit for a network **path**, and it is capped by the smallest MTU value that an IP packet encounters along its route.

When a host sends data across a network, one of the devices might use a smaller MTU than the sending host. When that situation happens, two things can occur:

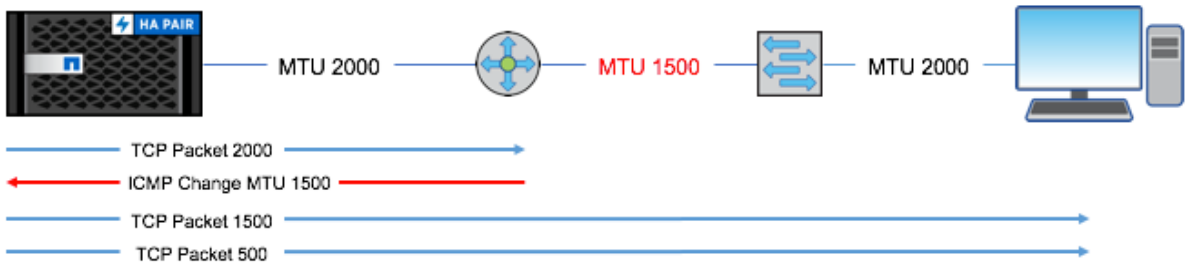
- If the sending host has not set the DF (Don't Fragment) bit in the packet header, the packet is fragmented.
- If the DF bit is set, the network device sends an ICMP packet back to the host, stating that the packet is too large and communicating its MTU value.

Path MTU discovery (PMTUD) is the process by which network endpoints share their maximum segment size in TCP connections to establish PMTU.

Note: When you use PMTUD, the DF bit is always set.

When a router receives a packet that is larger than its MTU, it checks the DF bit in the IP header. If the bit is not set, the router fragments the packet itself. If it is set, however, the router responds with an ICMP message back to the sending host, stating that the packet is too large and that it requires fragmentation. Figure 16 illustrates this scenario. The sending host, a NetApp FAS cluster, receives an ICMP message requesting that the source packet be split into two separate fragments before the router can send the traffic off to the upstream switch.

Figure 16) PMTUD between two network endpoints.



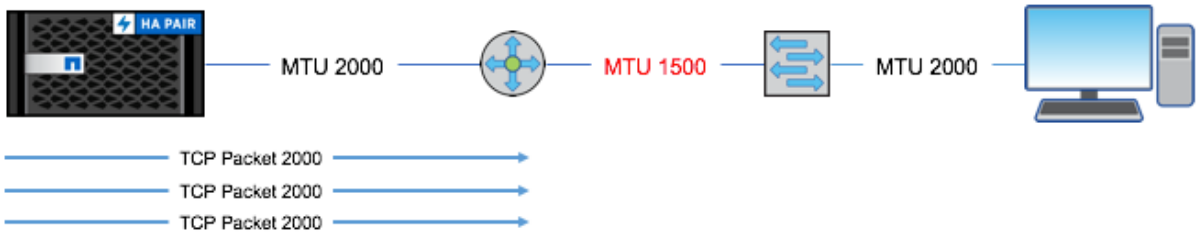
What Is a PMTU Black Hole?

A PMTU black hole is a scenario in which a network endpoint does not receive the ICMP message informing it to adjust its MTU. Thus the endpoint is unable to distinguish an MTU issue from network congestion or packet loss. This situation can occur for several reasons:

- Layer 2 network switches drop frames that are too large, and without ICMP capability (layer 3), there is no way to notify the sending host.
- Routers can have ICMP messages disabled. This action is frequently taken for security reasons in an effort to minimize exposing network topology.
- Firewalls can block ICMP messages.

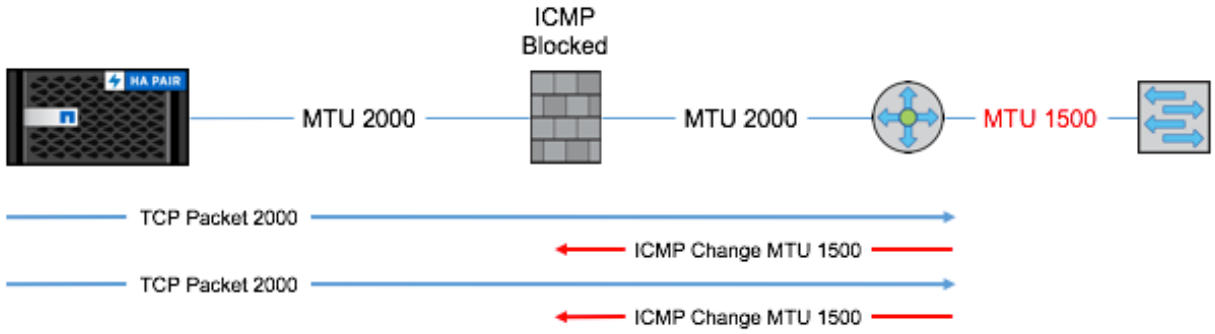
In Figure 17, our FAS is configured to use port MTU values that are larger than one of the supported connecting hops. In this scenario, the router is disallowed from sending ICMP traffic back to the host, preventing it from completing PMTUD. The FAS continues to send packets to the client, completely unaware that there is an MTU issue midstream. An administrator can easily mistake this behavior for any number of network issues that contribute to packet loss.

Figure 17) Router with ICMP disabled, preventing PMTUD.



In the scenario in Figure 18, a firewall is preventing ICMP traffic from reaching the FAS. Although our router is configured to send ICMP traffic back to the initiating host, the firewall is preventing that traffic from ever being received. Like the preceding scenario, PMTUD cannot be completed, and the sending host continues to send packets that are too large for our midstream network hop.

Figure 18) Firewall blocking ICMP messages, preventing PMTUD.



From the user standpoint, PMTU black holes can manifest themselves in several ways, including:

- Loss of connectivity, because the packet never reaches the destination
- Degradation of performance, as TCP congestion control algorithms kick in at the source

Eventually, after a certain number of retries, the source reduces its MTU and tries again. This attempt further affects the performance degradation that is seen under these conditions. Although data continues to flow, the next time that a large packet is sent, the scenario repeats itself.

PMTU Black Hole Detection and Resolution

Two network tools help in the detection of PMTU black holes:

- Network pings
- Packet traces

The `ping` command can help you troubleshoot by attempting to detect the network hop that is creating the PMTU black hole. By issuing pings of varying sizes, you can send packets between network endpoints with the DF bit set. From there, you can look for responding ICMP messages while checking the PMTU value, indicating where the black hole exists. This approach allows you to determine whether:

- Each network hop is configured to allow packets of the user-defined size.
- If a `Frag needed and DF set` message is returned to the user, the network hop was successfully able to send the ICMP response to the host for PMTUD.
- If a `Request timed out` message is returned to the user, ICMP might be blocked by any of the scenarios described previously.

```
localhost# ping -s 9000 -M do 10.1.1.1

localhost# From 10.1.1.1 icmp_seq=1 Frag needed and DF set (mtu = 9000)
localhost# From 10.1.1.1 icmp_seq=1 Frag needed and DF set (mtu = 9000)
localhost# From 10.1.1.1 icmp_seq=1 Frag needed and DF set (mtu = 9000)
localhost# From 10.1.1.1 icmp_seq=1 Frag needed and DF set (mtu = 9000)
```

Additionally, a network packet trace allows you to analyze these packets further, giving you the ability to look at all ingress ICMP packets for PMTUD negotiation messages.

When the black hole has been identified, reducing the MTU from the sending host to a supported value or modifying the MTU value of the network hop prevents these symptoms from causing further issues.

9 Use Cases

Many of the examples listed throughout this document are actual use cases that can be configured in a customer's environment. This section takes this information one step further and identifies more specific use cases. This section presents examples only, and they should be tuned to fit the configuration that is actually being implemented.

9.1 Creating Common Clustered Data ONTAP Network Objects

This section demonstrates the process of creating common NetApp clustered Data ONTAP network objects by using the CLI:

- IPspace:

```
ontaptme-rtp::*> network ipspace create -ipspace test-ipspace

ontaptme-rtp::*>
```

- Broadcast domain:

```
ontaptme-rtp::*> broadcast create -broadcast-domain testbroaddomain -ipspace test-
ipspace -mtu 1500 -ports ontaptme-rtp-01:e3b

  (network port broadcast-domain create)
```

Note: If you do not select an IPspace in the following command, the default IPspace is used. The ports selected cannot already be members of another broadcast domain; remove the ports that you want to use from their existing broadcast domains first.

If necessary, you can add ports to your newly created broadcast domain (`testbd`):

```
ontaptme-rtp::*> broadcast-domain add-ports -ipspace test-ipspace -broadcast-domain
testbroaddomain -ports ontaptme-rtp-02:e3b
ontaptme-rtp::*> broadcast-domain add-ports -ipspace test-ipspace -broadcast-domain
testbroaddomain -ports ontaptme-rtp-03:e3b
ontaptme-rtp::*> broadcast-domain add-ports -ipspace test-ipspace -broadcast-domain
testbroaddomain -ports ontaptme-rtp-04:e3b
```

- Failover groups:

One failover group was created when the broadcast domain (`testbd`) itself was created in the previous step. The following demonstrates creating a custom failover group named `testfg`, which resides in the broadcast domain along with the autogenerated one:

```
ontaptme-rtp::*> network interface failover-groups add-targets -vserver test -
failover-group testfg -targets ontaptme-rtp-02:e3b

ontaptme-rtp::*> network interface failover-groups show -vserver test
                                 Failover
Vserver          Group           Targets
---------------- --------------- --------------------------------------------
test
                 testbd
                                 ontaptme-rtp-02:e3b, ontaptme-rtp-03:e3b,
                                 ontaptme-rtp-01:a0a, ontaptme-rtp-01:a0a-10,
                                 ontaptme-rtp-01:a0a-20, ontaptme-rtp-01:e3b,
                                 ontaptme-rtp-04:e3b
                 testfg
                                 ontaptme-rtp-02:e3b, ontaptme-rtp-03:e3b,
                                 ontaptme-rtp-01:e3b, ontaptme-rtp-04:e3b
2 entries were displayed.
```



```
ontaptme-rtp::*>
```

Note: The autogenerated failover group cannot be modified. It always contains all the ports that belong to the broadcast domain. All other user-defined failover groups that are created in the same broadcast domain use subsets of the ports that belong to the autogenerated failover group.

- IFGRP:

The following example demonstrates creating an IFGRP, adding a physical port, and subsequently adding the IFGRP object into the broadcast domain named `testbd`:

```
ontaptme-rtp::*> ifgrp create ?
   [-node] <nodename>                              Node
   [-ifgrp] {<netport>|<ifgrp>}                    Interface Group Name
   [-distr-func] {mac|ip|sequential|port}          Distribution Function
   [-mode] {multimode|multimode_lacp|singlemode}   Create Policy

ontaptme-rtp::*> ifgrp create -node ontaptme-rtp-01 -ifgrp a0a -distr-func ip -mode
multimode_lacp

ontaptme-rtp::*> ifgrp add-port -node ontaptme-rtp-01 -ifgrp a0a -port e2b

ontaptme-rtp::*> ifgrp show
         Port       Distribution                   Active
Node     IfGrp      Function     MAC Address       Ports   Ports
-------- ---------- ------------ ----------------- ------- -------------------
ontaptme-rtp-01
         a0a        ip           02:a0:98:42:c2:f6 none    e2b

ontaptme-rtp::*> net port show -node ontaptme-rtp-01
  (network port show)
                                                            Speed (Mbps)
Node   Port      IPspace      Broadcast Domain Link   MTU    Admin/Oper
------ --------- ------------ ---------------- ----- ------- ------------
ontaptme-rtp-01
       a0a       Default      -                down     1500  auto/-
       e0M       Default      10.228.22.0/24_10.228.225.0/24
                                                 up       1500  auto/100
       e0a       Default      10.228.22.0/24_10.228.225.0/24
                                                 up       1500  auto/1000
       e0b       Default      10.228.22.0/24_10.228.225.0/24
                                                 up       1500  auto/1000
       e1a       Cluster      Cluster          up       9000  auto/10000
       e1b       Default      -                down     1500  auto/10
       e2a       Cluster      Cluster          up       9000  auto/10000
       e2b       Default      -                up       1500  auto/10000
       e3a       Default      -                up       1500  auto/10000
       e3b       test         testbd           up       1500  auto/10000
10 entries were displayed.

ontaptme-rtp::*> broadcast-domain add-ports -broadcast-domain testbd -ports ontaptme-
rtp-01:a0a -ipspace test
  (network port broadcast-domain add-ports)

ontaptme-rtp::*> net port show -node ontaptme-rtp-01
  (network port show)
                                                            Speed (Mbps)
Node   Port      IPspace      Broadcast Domain Link   MTU    Admin/Oper
------ --------- ------------ ---------------- ----- ------- ------------
ontaptme-rtp-01
       a0a       test         testbd           down     1500  auto/-
       e0M       Default      10.228.22.0/24_10.228.225.0/24
                                                 up       1500  auto/100
```



```
       e0a       Default      10.228.22.0/24_10.228.225.0/24
                                           up       1500  auto/1000
       e0b       Default      10.228.22.0/24_10.228.225.0/24
                                           up       1500  auto/1000
       e1a       Cluster      Cluster          up       9000  auto/10000
       e1b       Default      -                down     1500  auto/10
       e2a       Cluster      Cluster          up       9000  auto/10000
       e2b       Default      -                up       1500  auto/10000
       e3a       Default      -                up       1500  auto/10000
       e3b       test         testbd           up       1500  auto/10000
10 entries were displayed.

ontaptme-rtp::*>
```

- VLAN:

The following example creates two VLAN tagged ports (VLANs `10` and `20`) on IFGRP port `a0a`, which was created in the preceding example:

```
ontaptme-rtp::*> vlan create -node ontaptme-rtp-01 -vlan-name a0a-10

ontaptme-rtp::*> vlan show
                 Network Network
Node   VLAN Name Port    VLAN ID  MAC Address
------ --------- ------- -------- -----------------
ontaptme-rtp-01
       a0a-10    a0a     10       02:a0:98:42:c2:f6

ontaptme-rtp::*> vlan create -node ontaptme-rtp-01 -vlan-name a0a-20

ontaptme-rtp::*> vlan show
                 Network Network
Node   VLAN Name Port    VLAN ID  MAC Address
------ --------- ------- -------- -----------------
ontaptme-rtp-01
       a0a-10    a0a     10       02:a0:98:42:c2:f6
       a0a-20    a0a     20       02:a0:98:42:c2:f6
2 entries were displayed.

ontaptme-rtp::*>
```

Add the newly created VLANs to the broadcast domain `testbd`:

```
ontaptme-rtp::*> broadcast-domain show -ipspace test
  (network port broadcast-domain show)
IPspace Broadcast                                       Update
Name    Domain Name    MTU  Port List                   Status Details
------- ----------- ------  ---------------------------- --------------
test    testbd        1500
                            ontaptme-rtp-01:e3b          complete
                            ontaptme-rtp-02:e3b          complete
                            ontaptme-rtp-03:e3b          complete
                            ontaptme-rtp-04:e3b          complete
                            ontaptme-rtp-01:a0a          complete

ontaptme-rtp::*> broadcast-domain add-ports -ipspace test -broadcast-domain testbd -
ports ontaptme-rtp-01:a0a-10,ontaptme-rtp-01:a0a-20
  (network port broadcast-domain add-ports)

ontaptme-rtp::*> broadcast-domain show -ipspace test
  (network port broadcast-domain show)
IPspace Broadcast                                       Update
```



```
Name    Domain Name    MTU  Port List                       Status Details
------- ----------- ------  ------------------------------ --------------
test    testbd        1500
                            ontaptme-rtp-01:e3b            complete
                            ontaptme-rtp-02:e3b            complete
                            ontaptme-rtp-03:e3b            complete
                            ontaptme-rtp-04:e3b            complete
                            ontaptme-rtp-01:a0a            complete
                            ontaptme-rtp-01:a0a-10         complete
                            ontaptme-rtp-01:a0a-20         complete

ontaptme-rtp::*>
```

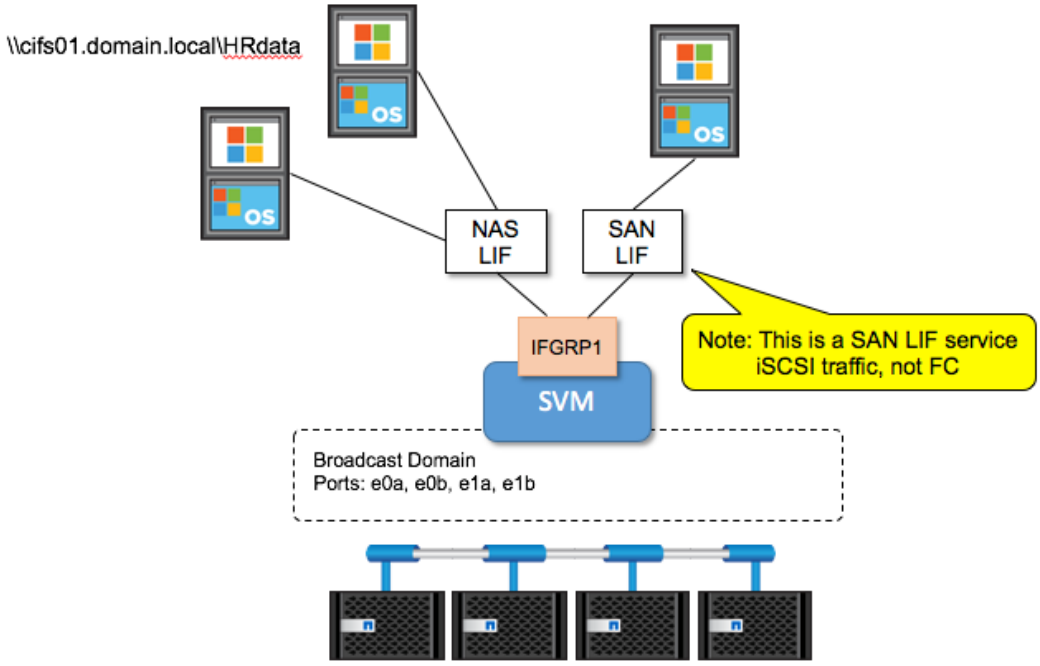
- Interface:

```
ontaptme-rtp::*> net int create -vserver test -lif syslif1 -role intercluster -home-
node ontaptme-rtp-01 -home-port e3b -address 192.168.1.10 -netmask 255.255.255.0 -
status-admin up -failover-policy local-only -firewall-policy intercluster -auto-revert
false -allow-lb-migrate false -failover-group testbd
  (network interface create)

ontaptme-rtp::*> net int show -vserver test -instance
  (network interface show)

                    Vserver Name: test
           Logical Interface Name: syslif1
                             Role: intercluster
                    Data Protocol: -
                        Home Node: ontaptme-rtp-01
                        Home Port: e3b
                     Current Node: ontaptme-rtp-01
                     Current Port: e3b
                Operational Status: up
                   Extended Status: -
                        Numeric ID: 1047
                           Is Home: true
                   Network Address: 192.168.1.10
                           Netmask: 255.255.255.0
              Bits in the Netmask: 24
                  IPv4 Link Local: -
                      Subnet Name: -
            Administrative Status: up
                  Failover Policy: local-only
                  Firewall Policy: intercluster
                      Auto Revert: false
                      Sticky Flag: false
    Fully Qualified DNS Zone Name: none
          DNS Query Listen Enable: false
   Load Balancing Migrate Allowed: false
             Load Balanced Weight: load
              Failover Group Name: testbd
                         FCP WWPN: -
                   Address family: ipv4
                          Comment: -
                  IPspace of LIF: test
   Is Dynamic DNS Update Enabled?: -

ontaptme-rtp::*>
```


9.2 Configuration That Shows Cohosting of SAN/NAS LIFs

Figure 199 displays an example configuration in which a NAS LIF and a SAN LIF are both serving requests from the same IFGRP port.

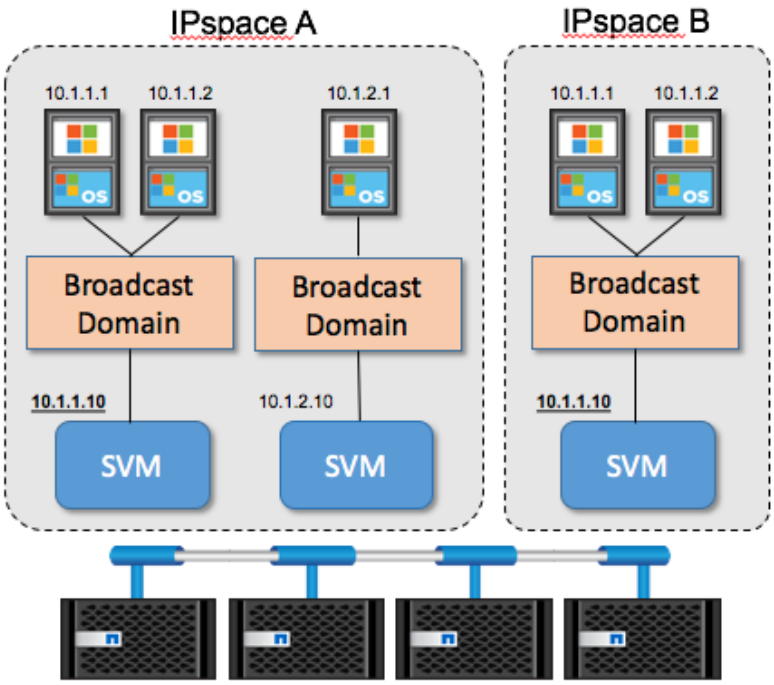
Figure 19) SAN/NAS configuration: SVM that uses ports to serve NAS, iSCSI, and traffic.



9.3 Multiple IPspaces with Overlapping IP Addresses

In Figure 20, notice that the 10.1.1.10 address is used for accessing two different SVMs in two different IPspaces. This functionality isn't necessarily new; we have seen this type of configuration with our 7-Mode configurations in the past. However, it is new to clustered Data ONTAP beginning with 8.3.

Figure 20) Multiple IPspaces with the same IPs assigned to an entry interface: overlapping IP addresses.



9.4 SVM IPspace Placement After Upgrading from Data ONTAP 8.2 to 8.3

This use case covers the remediation steps that are required after you upgrade from clustered Data ONTAP 8.2 to 8.3. When an SVM is placed into the default IPspace after the upgrade but network connectivity requires that it be placed into a separate IPspace, you must make manual configurations. This situation might involve some up-front planning, because you are coming from an 8.2 environment that isn't aware of objects such as broadcast domains.

Because moving an SVM from one IPspace to another is unsupported, to migrate the data to the appropriate IPspace, you must execute the following steps:

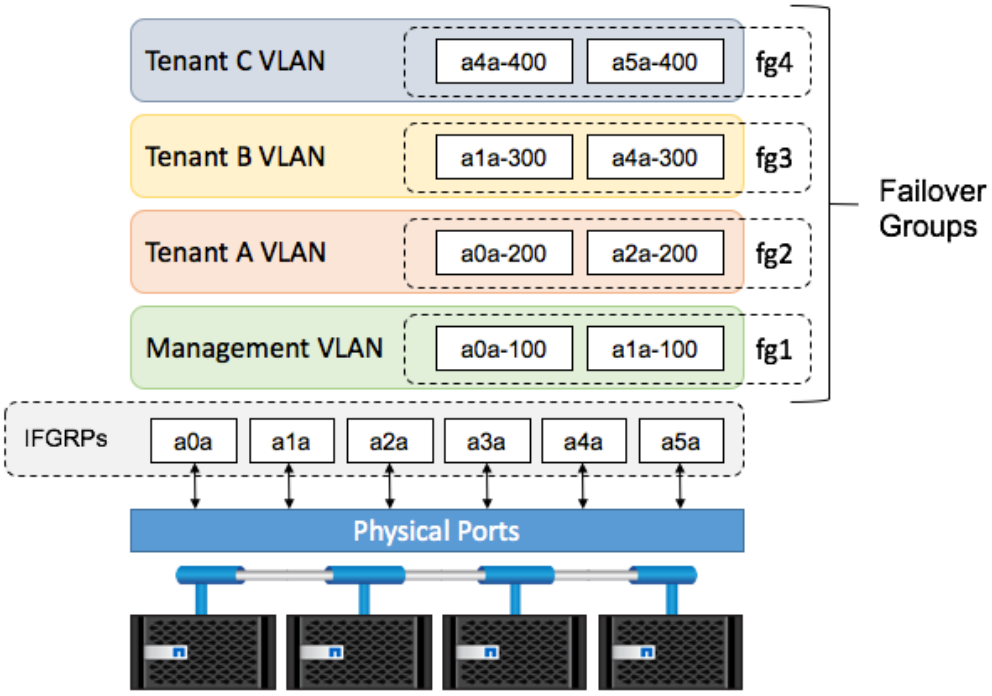
1. Create the custom IPspace.
2. Create the custom broadcast domain, which also creates your failover group. Assign ports correctly when you create the broadcast domain; otherwise, failure to connect to services and loss of connectivity from services could occur.
3. Create the appropriate LIFs for data connectivity. This step includes the intercluster LIFs that step 4 uses to replicate the data from the SVM in the default IPspace to the SVM in the custom IPspace.
4. Use NetApp SnapMirror to create an SVM peering relationship and to copy the data from one volume to the other.

Note: In the enhancements that are planned for this consideration in a future release, upgrading to an IPspace-aware clustered Data ONTAP release will not involve manual configurations.

9.5 Availability and Performance Best Practices for Clustered Data ONTAP 8.2

Different LIF/failover group/VLAN/IFGRP configurations are possible in a clustered Data ONTAP environment. The recommended best practice is to use the configuration in Figure 21. This configuration takes advantage of the clusterwide failover capabilities of failover groups, the port aggregation functionality of IFGRPs, and the security aspects of VLANs. For more examples, see the Clustered Data ONTAP 8.2 Network Management Guide.

Figure 21) Example configuration that uses IFGRPs, failover groups, and VLANs.



Additional Resources

- Clustered Data ONTAP Networking Guides
- TR-4191: Best Practices Guide for Clustered Data ONTAP 8.2.x and 8.3.x Windows File Services
- TR-4067: Clustered Data ONTAP NFS Implementation Guide
- TR-4080: Scalable SAN Best Practices in Data ONTAP 8.3

Contact Us

Let us know how we can improve this technical report.

Contact us at docfeedback@netapp.com.

Include TECHNICAL REPORT 4182 in the subject line.

Refer to the Interoperability Matrix Tool (IMT) on the NetApp Support site to validate that the exact product and feature versions described in this document are supported for your specific environment. The NetApp IMT defines the product components and versions that can be used to construct configurations that are supported by NetApp. Specific results depend on each customer's installation in accordance with published specifications.