lenovo

Lenovo Networking Best Practices for Configuration and Installation

Benefit from the expansive knowledge of Lenovo Networking experts

Discover design strategies to maximize network performance

Learn about the latest switching and routing features

Implement switch security and management features

Scott Irwin
Scott Lorditch
Ted McDaniel
William Nelson
Matt Slavin
Megan Gilge

Lenovo Networking Best Practices for Configuration and Installation

August 2015

SG24-8245-00

Note: Before using this information and the product it supports, read the information in “Notices” on page vii.

Last update on August 2015

Contents

Notices

Lenovo may not offer the products, services, or features discussed in this document in all countries. Consult your local Lenovo representative for information on the products and services currently available in your area. Any reference to a Lenovo product, program, or service is not intended to state or imply that only that Lenovo product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any Lenovo intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any other product, program, or service.

Lenovo may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

Lenovo (United States), Inc.
1009 Think Place - Building One
Morrisville, NC 27560
U.S.A.
Attention: Lenovo Director of Licensing

LENOVO PROVIDES THIS PUBLICATION “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. Lenovo may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

The products described in this document are not intended for use in implantation or other life support applications where malfunction may result in injury or death to persons. The information contained in this document does not affect or change Lenovo product specifications or warranties. Nothing in this document shall operate as an express or implied license or indemnity under the intellectual property rights of Lenovo or third parties. All information contained in this document was obtained in specific environments and is presented as an illustration. The result obtained in other operating environments may vary.

Lenovo may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any references in this publication to non-Lenovo Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this Lenovo product, and use of those Web sites is at your own risk.

Any performance data contained herein was determined in a controlled environment. Therefore, the result obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Trademarks

Lenovo, the Lenovo logo, and For Those Who Do are trademarks or registered trademarks of Lenovo in the United States, other countries, or both. These and other Lenovo trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by Lenovo at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of Lenovo trademarks is available on the Web at http://www.lenovo.com/legal/copytrade.html.

The following terms are trademarks of Lenovo in the United States, other countries, or both:

Blade Network Technologies®	Flex System™	Lenovo(logo)®
BladeCenter®	Lenovo®	vNIC™
BNT®	RackSwitch™	xSeries®

The following terms are trademarks of other companies:

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Preface

Networking is the foundation of any multisystem solution and this document provides you with the tools to enable a successful implementation. The features that are available on the Lenovo Networking switches can be overwhelming at first; however, you can construct it easily if the configuration is broken down into its basic building blocks.

Each chapter in this book focuses on the key areas of the switch configuration, starting with the network design and proceeding through the network layer features, security, and server integration. Details are provided about how to configure the switch and to assure that it is functioning properly. This publication does not provide a complete description of every switch feature but instead focuses on the topics that are generally considered key to a successful implementation.

This publication is targeted towards technical professionals (networking team, consultants, technical support staff, and IT specialists) who are responsible for supporting the integration of systems into new and existing networks.

The team who wrote this book

This book was produced by a team of Lenovo Networking specialists:

Scott Irwin is a Consulting Systems Engineer (CSE) with Lenovo Networking, formerly from IBM and Blade Network Technologies® (BNT®). His networking background spans well over 16 years as a Customer Support Escalation Engineer and a Customer-facing Field Systems Engineer. His focus is on deep customer troubleshooting and his responsibilities include supporting customer proof of concepts, assistance with paid installations and training, and supporting pre- and post-sales activities with customers in the Public Sector, High Frequency Trading, Service Provider, Midmarket, and Enterprise markets.

Scott Lorditch is a World-Wide Consulting Systems Engineer for Lenovo Networking based out of greater Denver, Colorado. He provides network consulting and training skills to Lenovo Networking and EBG teams throughout the world. His background includes network design and operations for a major global bank, a global soft-drink company, and a telecommunications carrier. Scott was one of the original employees of Blade Network Technologies before its acquisition by IBM. He joined Lenovo through the acquisition of the IBM System x team.

Ted McDaniel is a Senior Product Engineer on the Lenovo System x Product Engineering team in Morrisville, NC. His current focus is on all aspects of Flex System and BladeCenter® networking: from the operating system to the NICs and the chassis switches. He started supporting the all aspects of the BladeCenter chassis six months after it released, and has worked on the Flex System Enterprise Chassis since it was released.

William Nelson is a Worldwide Technical Sales Leader for Lenovo Networking. He joined Lenovo from IBM and BNT and has over 30 years of computing networking experience. He was one of the four founders of BNT and an early member of Centillion Networks and Alteon Web Systems. He is an evangelist for Lenovo's Networking products to the sales and customer communities and is the key voice of the customer to the networking product and engineering teams.

Matt Slavin is a Consulting Systems Engineer for Lenovo Networking, based out of Tulsa, Oklahoma. He provides network consulting skills to the Americas. He has over 30 years of hands-on systems and network design, installation, and troubleshooting experience. Most recently, he has focused on data center networking, where he is leading client efforts to adopt new technologies into day-to-day operations. Matt joined Lenovo through the acquisition of the IBM System x team. Before that acquisition, he worked at some of the top systems and networking companies in the world.

Megan Gilge is a Project Leader in the IBM Redbooks organization. Before joining the Redbooks team three years ago, she was an Information Developer in the IBM Semiconductor Solutions and User Technologies areas for eight years.

Thanks to the following people for their contributions to this project:

Jon Tate
IBM Redbooks

Tim Shaughnessy
Lenovo Networking

David Watts
Lenovo Press

Comments welcome

Your comments are important to us!

We want our books to be as helpful as possible. Send us your comments about this book or in one of the following ways:

- Use the online **Contact us** review Redbooks form found at:

  ibm.com/redbooks
- Send your comments in an email to:

  redbooks@us.ibm.com

1

Introduction

The network is a key element of a multi-system solution and this Lenovo Press publication describes the key elements that must be considered when the Lenovo Networking switches are integrated into a new or existing network.

This chapter includes the following topics:

These topics are basic building blocks of the network switch configuration and must be considered equally important when determining your strategy for designing and configuring your networking solution.

1.1 Network design strategy

When the networking solution is designed, the following key strategies can lead to successful designs:

- Designs should be as simple as possible. Push more complex configuration elements (such as routing) to core or aggregation layer switches to minimize the configuration of the larger number of access layer switches. This configuration reduces design, implementation, and troubleshooting efforts.
- Symmetrical configurations can minimize errors. The more you can keep the configuration of the switches at each layer as similar as possible, the easier it is to manage the devices. This configuration also reduces the possibility of errors when the new network nodes are rolled out. Many times, the only difference that is required between switch configurations is the host names and IP addresses.

1.2 Connecting the switches to the network and access nodes

Success in integrating the new networking switches into a new or existing environment requires a full understanding of the following components:

- Mission of the solution

  Determine whether network redundancy, low latency, or expanded bandwidth (among other items) are required.
- Existing infrastructure

  Determine the Layer 1 (physical, such as 1GBaseT, 1GBaseSR, and 10GBaseSR), 2 (data link, such as VLANs, LAGs, and STP) and 3 (network, such as IP routing) requirements.
- Management requirements

  Determine whether there are remote authentication (TACACS+, and so on) and management system (SNMPv2c, SNMPv3, and so on) requirements.
- Converged network requirements

  Determine whether FCoE or any other network-attached storage is used.

All of these considerations are important to understand. More information is provided in this document to aid with the design and implementation of these features.

1.3 Lenovo networking switches

The Lenovo networking switches are a family of RackSwitch™ top-of-track (TOR) switches and embedded switches for the Flex System and BladeCenter chassis. These switches provide a full range of Layer 2 and 3 features along with the associated management features. For more information about the Lenovo switches, see this website:

http://www.lenovo.com/networking

For more information about a public community on the Lenovo website where you can ask questions and get more information about the switches, see this website:

https://forums.lenovo.com/t5/Enterprise-Networking/bd-p/nw01_eg

1.4 About this book

The topics that are covered in this publication are applicable to the Lenovo Networking switches, including the RackSwitch, Flex System, and BladeCenter switch families. Each chapter covers key subjects to be considered in the design and implementation of the network. This book is broken down into several chapters that can be classified into the following categories:

- Designing the network: Chapter 2, “Network design and topologies” on page 5
- Connecting to the network:
  - Chapter 3, “Layer 1 technologies” on page 33
  - Chapter 4, “Layer 2 technologies” on page 47
  - Chapter 5, “Layer 3 technologies” on page 99
- Managing the network:
  - Chapter 6, “Securing access to the switch” on page 115
  - Chapter 7, “Operation and management” on page 125
- Converged networks: Chapter 8, “Converged networking” on page 187
- Connecting end nodes/hosts: Chapter 9, “Integrating with hosts” on page 193

2

Network design and topologies

This chapter describes several network topologies that are frequently used by customers and can be used for future deployments. The topologies that are presented focus on networks that have one or more Flex System™ chassis that connect to an upstream network. The chassis are shown with embedded switches included within them. However, these topologies can also be used with rack-mounted servers and a pair of top-of-rack switches.

This chapter includes the following topics:

2.1 Sample topologies

The presentation of each topology includes information about its merits and constraints, and about when it is most appropriate to use that topology. Any of the topologies that are shown can also be extended to include Fibre Channel over Ethernet (FCoE). For more information about FCoE, see 2.2.1, “FCoE with vLAG” on page 16.

Most of the examples that are shown can be implemented by using a feature that is called *Easy Connect*. Several of the examples in this chapter show the addition of an Easy Connect element. For more information about what Easy Connect represents and its various iterations and interactions, see 2.2.4, “Easy Connect” on page 28.

In general, the criteria for selecting the topology includes the following items:

- The capabilities of the devices in the customer’s network, which are immediately upstream from the embedded switches. This criterion includes the bandwidth of the network (1 Gb versus 10 Gb).
- The customer’s standards and practices for network interface card (NIC) teaming on their servers.
- The customer’s preferences for stacking and management of the embedded switches.

2.1.1 Full mesh topology with Virtual Link Aggregation

This topology is preferred (except for when conditions prevent its use, such as those conditions that described later in this section). It provides connectivity with the following qualities:

- High availability, which enables the environment to survive the failure of one of two embedded switch modules, one of two upstream switches (or the links that connect to them), or both.
- All of the links between an embedded switch in the Flex System chassis and an upstream switch are active and can carry production traffic. None of the links are blocked by Spanning Tree to prevent network loops.
- The server-facing (INT*x*) ports on the embedded switches can be channeled together. This configuration works with the high availability features. Aggregation-based Active/active NIC teaming modes are available if the server’s ports are channeled.

Do not implement this topology if any of the following conditions are true:

- The upstream switches do not support a form of cross-switch link aggregation, such as vPC, VSS, Virtual Link Aggregation (vLAG), MCLAG, or a stacking feature. In this case, other designs should be used, and can enable most of the availability and usage features for this topology. Those other designs are described in 2.1.3, “Stacking: Full mesh” on page 9 and 2.1.4, “Flex System Interconnect Fabric” on page 12.
- The customer does not have two switches that they plan to use to connect to the configuration. A single upstream switch design is not recommended because it contains a single point of failure and can isolate the entire chassis from the remainder of the network if that single switch fails.

This topology with a single Flex System chassis is shown in Figure 2-1 on page 7. Equivalent designs can be deployed by using BladeCenter chassis or by using rack-mounted servers that are dual-homed to a pair of top-of-rack switches.

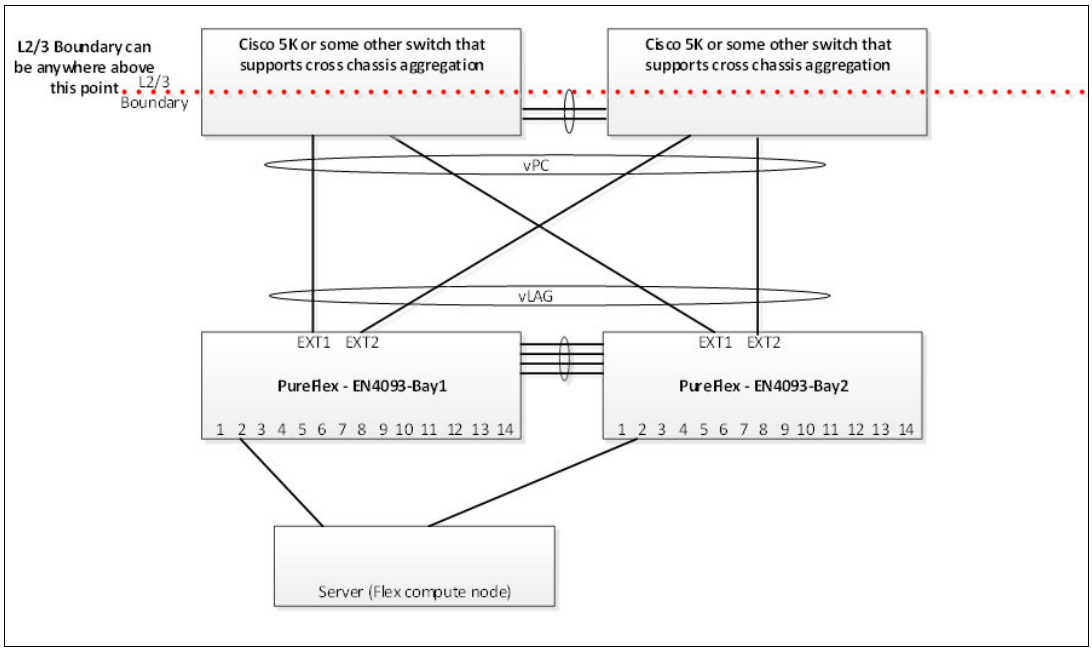


Figure 2-1 Full mesh network design

Key portions of the Lenovo switch configuration for this topology are shown in Figure 2-2. This configuration fragment shows the portions of the overall configuration that relate to vLAG.

```
vlag enable
vlag tier-id 50
vlag isl adminkey 5152
vlag adminkey 102 enable
vlag adminkey 304 enable
vlag hlthchk peer-ip 10.1.1.2
```

Figure 2-2 Configuration commands for vLAG

Specific information for vLAG

When you are using vLAG in this topology or elsewhere, it is important to consider the requirements for a successful vLAG deployment. Such factors as proper sizing of the Interswitch Link (ISL), configuration of a health check network, and so on, are critical for a healthy vLAG environment. For more information about vLAG and its requirements, see 4.1, “Virtual Link Aggregation Group considerations” on page 48.

2.1.2 Inverted U topology with failover

This configuration is an equivalent for Ethernet to the common practice in SAN designs of having two parallel networks from the adapter, which is host bus adapter (HBA) for Fibre Channel, to the storage device. This configuration is often referred to as a *SAN-A/SAN-B* design. This design has a “left side” network that is tied to one NIC port on the server, flowing through the left embedded switch within the Flex System chassis, and connecting up to the left aggregation/core switch in the customer environment. The second NIC port similarly connects through the “right side” network.

Because there are two distinct switching devices at every tier, this topology cannot support active/active NIC teaming modes that are switch dependent (aggregation). It supports, and should be used with, switch independent mode active/active teaming (for example, Linux bonding mode 5 or VMware ESX default teaming mode, route based on originating virtual port ID), and all active/standby teaming modes such as Linux bonding mode 1. For more information about teaming modes and their ramifications, see *NIC Virtualization in Flex System Fabric Solutions*, SG24-8223, which is available at this website:

http://lenovopress.com/sg248223

This topology provides connectivity to the upstream network with the following qualities:

- High availability is provided in that the environment survives the failure of a single embedded switch or a single upstream switch.
- For servers in which NIC teaming or bonding is not used, this configuration does not provide L2 high availability. Serves that are not using NIC teaming or bonding can use other tools, such as local routing, to achieve high availability; however, in general, this architecture is most commonly used without teaming or bonding on the servers.
- Assuming the server is using teaming or bonding, the *failover* feature is required when this design is used and must be explicitly configured. The failover feature administratively disables server-facing (internal) ports when the external ports that connect the switch to its upstream neighbor fail. Without the failover feature, if the uplinks out of the switch failed, the server is unaware of this failure and continues to send traffic to the switch, which drops traffic when there is no upstream path available.

You might want to choose another design for the following reasons:

- High availability support for servers that are not configured with some form of NIC teaming (bonding) is not available. Other designs provide more robustness for servers that are configured in such a manner.
- You want a stacking or fabric design that spans multiple chassis.
- The customer has only one aggregation or core switch (avoid a single upstream switch design because it represents a single point of failure).

This topology with a single Flex System chassis is shown in Figure 2-3. Equivalent network designs can be deployed with a BladeCenter chassis or by using rack-mount servers that are dual-homed to a pair of top-of-rack switches.

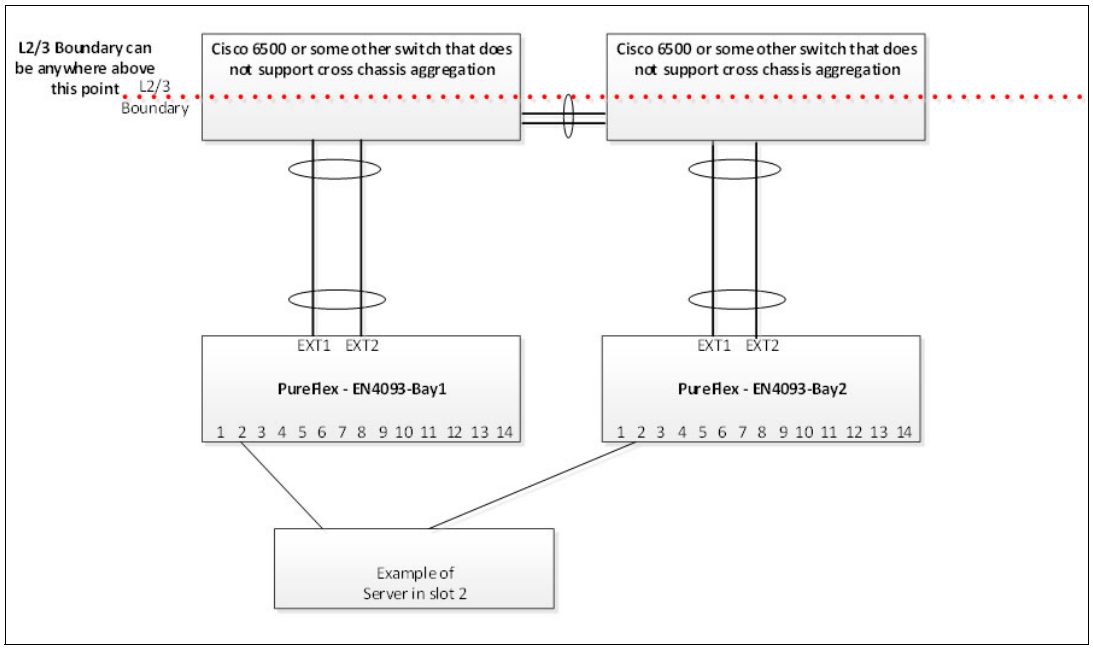


Figure 2-3 Inverted U network design

The *failover* feature, which is wanted when this topology is used, is configured to specify which external facing ports are used as uplinks and which server-facing ports are administratively disabled when the uplinks are down. It can operate on a per-VLAN basis or on a per-switch basis, meaning that only server-facing ports on the same VLANs as the failed uplinks are disabled if so configured. An example configuration of the *failover* feature is shown in Figure 2-4. More options for failover are also available by using the manual monitor (`mmon`) mode to provide greater flexibility in the configuration.

```
failover enable
failover trigger 1 amon admin-key 102
failover trigger 1 enable
```

Figure 2-4 Configuration commands for failover with auto-monitor tracking an LACP uplink key

2.1.3 Stacking: Full mesh

Stacking is supported on many of the 10 Gb products from Lenovo Networking; for example, the CN4093, EN4093, G8264, G8264CS, and the BladeCenter Virtual Fabric 10 Gb switch. The SI4093 module does not support stacking because it is not a fully functional switch; however, it can be a tributary in a Flex System Interconnect Fabric topology, which is described in 2.1.4, “Flex System Interconnect Fabric” on page 12.

Stacking support in Lenovo Networking products allows several switches to be configured and managed as a single unit, which reduces the number of network devices that must be managed. The stacked devices also share their forwarding databases (FDB), which enables link aggregation from ports that are on different members of the stack (similar to the capability offered by vLAG) for both server-facing and externally facing ports.

Embedded switches can be stacked across several chassis because a stack of EN4093 switches can have up to eight member switches.

When switches are run in stacked mode, the following constraints apply:

- Firmware updates require a reboot of the entire stack to take effect. If a host has redundant links that are connected to a single stack, this stack reload should be done only during a maintenance window where it is accepted that there are instances in which all of the servers that are connected to the stack are unreachable.
- On the current firmware releases on most Lenovo stacked switches, stacking does not have *local preference,* which is provided with vLAG. Lack of local preference means traffic that must use an aggregation across the stack to get out might use the stacking links to a member switch in the stack instead of a local uplink that is available (inefficient use of stacking links and increased packet latency).
- Several features are unavailable in stack mode, including dynamic L3 forwarding (routing) and sFlow support. The application guide for each model includes a current list of features, which changes from release to release.
- Stacking requires that all members of the stack are the same model, except that EN4093 and CN4093 embedded switches can be stacked together, with no more than two CN4093 switches in the stack.

Guidelines for stacking

Adhere to the following stacking guidelines:

- There are two primary configurations for stacking when switches that are embedded in Pure Flex chassis (4093 class) are used:
  - A single stack across a maximum of four chassis.
  - A two-stack design in which one uses bay 1 in multiple chassis and a second stack uses bay 2 in the same chassis.

  The use of one stack causes a brief but complete outage when the stack is rebooted, whether for a firmware upgrade or for any other reason. The use of two stacks does not suffer this outage (each stack can be reloaded independently), but causes the loss of the advantages of support for aggregation to hosts connected to the two separate stacks.
- Stacks always have a single master switch in control of the stack. It is a preferred practice to also have a backup master, which must be explicitly configured. If the stack uses embedded switches and spans multiple chassis, it is preferred to have the master and the backup be in different chassis.

For more information about stacking, see 4.2, “Stacking” on page 60.

Figure 2-5 shows a stacking configuration with three Flex System chassis.

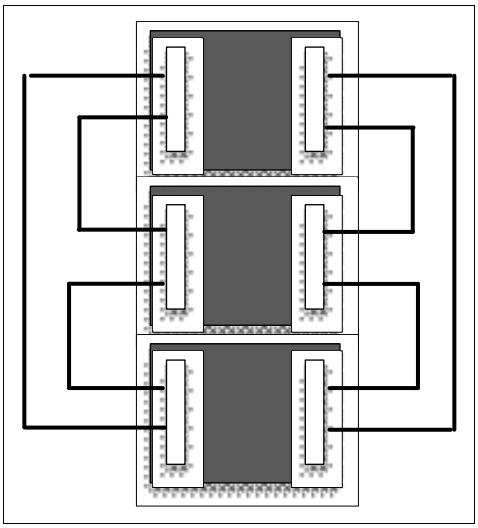


Figure 2-5 Three chassis with two independent stacks

Figure 2-6 shows three chassis with a single stack.

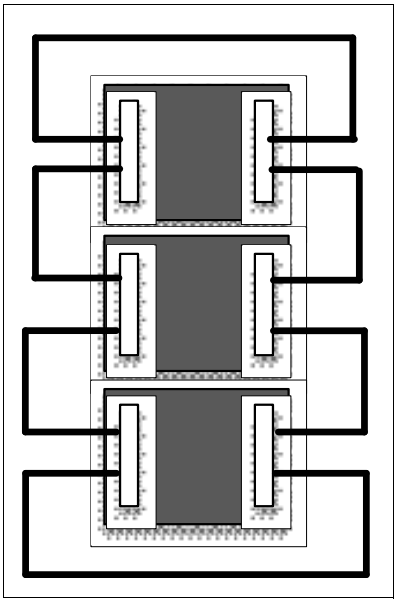


Figure 2-6 Three chassis with a single stack

Figure 2-7 shows configuration commands for stacking. For more information about these commands, see the *Application Guide* and the *Command Reference* manuals for the various products. In general, the stacking VLAN is left at the default value of 4090 and so that command is used infrequently.

```
boot stack enable
boot stack mode master
boot stack higig-trunk EXT9,EXT10
boot stack vlan 4090
```

Figure 2-7 Configuration commands for stacking (reload required to take effect)

2.1.4 Flex System Interconnect Fabric

Flex System Interconnect Fabric (sometimes referred to as Flex Fabric or FSIF) is a technology that is similar to stacking. It consists of a pair of G8264CS switches that act as aggregation switches to a group of SI4093 access switches (up to nine chassis with two SI4093 modules in each Flex System chassis). For more information about the features of the Flex Fabric, see *Flex System Interconnect Fabric*, TIPS1183, which is available at this website:

http://lenovopress.com/tips1183

The following benefits of Flex Fabric are similar to the benefits of stacking, without some of the limitations:

- The entire fabric has a single configuration and is managed as a single switching entity.
- Link aggregation to a pair of upstream switches from uplink ports on the G8264CS switches is easy to configure and does not require a vLAG configuration.
- Link aggregation between ports to the same server from each of the two SI4093 modules in a chassis is similarly easy to configure without vLAG. It enables active/active aggregation-based NIC teaming modes on the servers.
- Unlike stacking, Flex Fabric supports an automated staggered firmware upgrade process, which reboots elements of the fabric in a predefined sequence. This process allows a firmware upgrade to be done without losing reachability to the servers.

You might not want to use Flex Fabric for the following reasons:

- Unless enough chassis are included in the fabric immediately or in a planned expansion of the fabric environment, the cost of the pair of G8264CS switches can make this design unappealing.
- There are features that are not supported on Flex Fabric. Most notably, dynamic Layer-3 routing is not supported, and there is no support for Spanning Tree because loops cannot be created with Flex Fabric.

The ports on the switching elements in a fabric feature the following restrictions in their roles:

- The external ports on the SI4093 modules are set by default to be fabric ports and are usable only for upstream connections to the G8264CS switch.
- Ports 17-35 on each G8264CS switch are set by default to be fabric ports, which can be used only to connect to the SI4093 modules.
- The remaining ports on the G8264CS switch are used as uplinks to the customer's network or for use as FC ports. The Ethernet uplink configuration can be as a single link aggregation that uses *link aggregation* between the upstream switches or uses *hotlinks* to fail over between upstream switches that cannot support cross-switch PortChannels.
- Unused ports on SI4093 modules or G8264CS switches in Flex Fabric cannot be used to attach rack-mounted servers to the fabric. This restriction also precludes attaching a freestanding packet capture device to any of those unused ports.

Figure 2-8 shows a Flex System Interconnect Fabric that is connected to upstream switches with link aggregation.

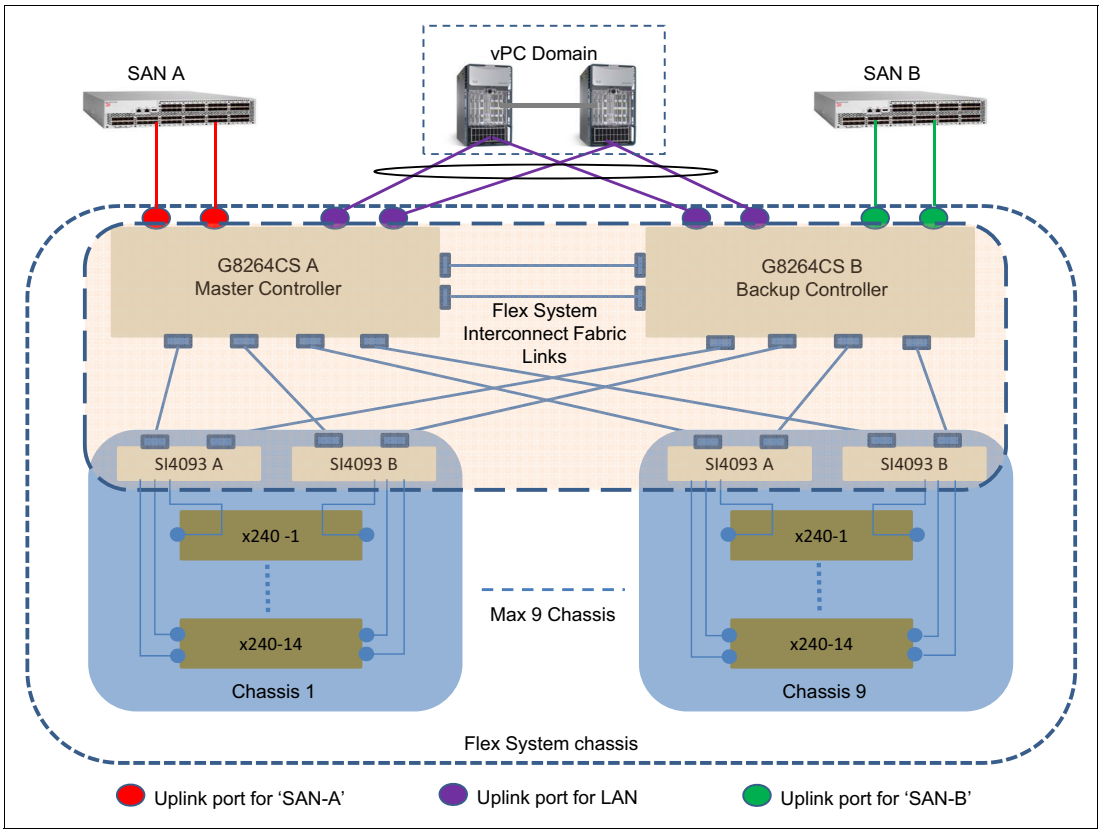


Figure 2-8 System Interconnect Fabric connected to upstream switches with link aggregation

Configuration excerpts for Flex Fabric (in *link aggregation* mode and in *hotlinks* mode) are shown in Figure 2-8. These excerpts show only the ports that are used to connect to the upstream network and do not show the assignment of VLANs to these uplink ports.

The two G8264CS switches in a Flex Fabric are always assigned switch numbers 1 and 2.

Example 2-1 shows Flex Fabric with static link aggregation. This configuration unconditionally aggregates two ports on each of the G8264CS switches. Connect it to a static vLAG or vPC instance on the upstream switches.

Example 2-1 Flex Fabric with static link aggregation

```
portchannel 1 port 1:1,1:2,2:1,2:2 enable
```

Example 2-2 shows Flex Fabric with Dynamic Link Aggregation (LACP). It uses LACP to safely aggregate two ports on each of the G8264CS switches. Connect it to an LACP instance of vLAG or vPC on the upstream switches.

Example 2-2 Flex Fabric with Dynamic Link Aggregation (LACP)

```
interface port 1:1,1:2,2:1,2:2
lacp key 1212
lacp mode active
```

Example 2-3 shows Flex Fabric with *hotlinks*. This configuration uses one port on each G8264CS switch to actively carry traffic to one of the upstream switches and have another port on each G8264CS switch standing by to carry traffic if the first upstream device suffers a failure. This example assumes that there is a static two-PortChannel that is configured on each of the two upstream devices.

Example 2-3 Flex Fabric with hotlinks

```
portchannel 1 port 1:1,2:1 enable
portchannel 2 port 1:2,2:2 enable
hotlinks trigger 1 master porchannel 1
hotlinks trigger 1 backup portchannel 2
hotlinks trigger 1 enable
hotlinks enable
```

Example 2-4 shows Flex Fabric with hotlinks and LACP. This configuration assumes that the two upstream switches each have an LACP two-PortChannel configured.

Example 2-4 Flex Fabric with hotlinks and LACP

```
interface port 1:1,2:1
lacp key 1111
lacp mode active
interface port 1:2,2:2
lacp key 1212
lacp mode active
hotlinks trigger 1 master adminkey 1111
hotlinks trigger 1 backup adminkey 1212
hotlinks trigger 1 enable
hotlinks enable
```

2.1.5 Traditional STP design with blocking

This topology was commonly used when functions, such as those provided by vLAG and stacking, were not available. It uses a partial mesh between the embedded (or server adjacent) switches and two upstream switches that are cross-connected to each other. The loops, which are built in to this design, are blocked by STP, which puts some ports into a blocking status to prevent a broadcast storm. Operationally, this design resembles an inverted-U topology; however, the blocked links can take over if there are switch or link failures.

The major drawback of this design is that it does require the use of STP and results in wasted bandwidth owing to blocked links. For more information about STP, including the multiple versions of the STP protocol, see 4.9, “Spanning Tree Protocol” on page 78.

Because ports that are blocked by STP do not carry production traffic, sufficient bandwidth must be built into the topology to carry the expected loads with these ports idle. This topology uses the available links inefficiently. In some cases, all of the available links can be used by setting STP parameters (link cost and switch priority) in such a way that some links are blocked for about half of the VLANs in use and other links are blocked for the remaining VLANs.

A network that uses the Traditional STP design on a Flex chassis is shown in Figure 2-9. Equivalent designs can be deployed with a BladeCenter chassis or with rack-mounted servers dual-homed to a pair of top-of-rack switches.

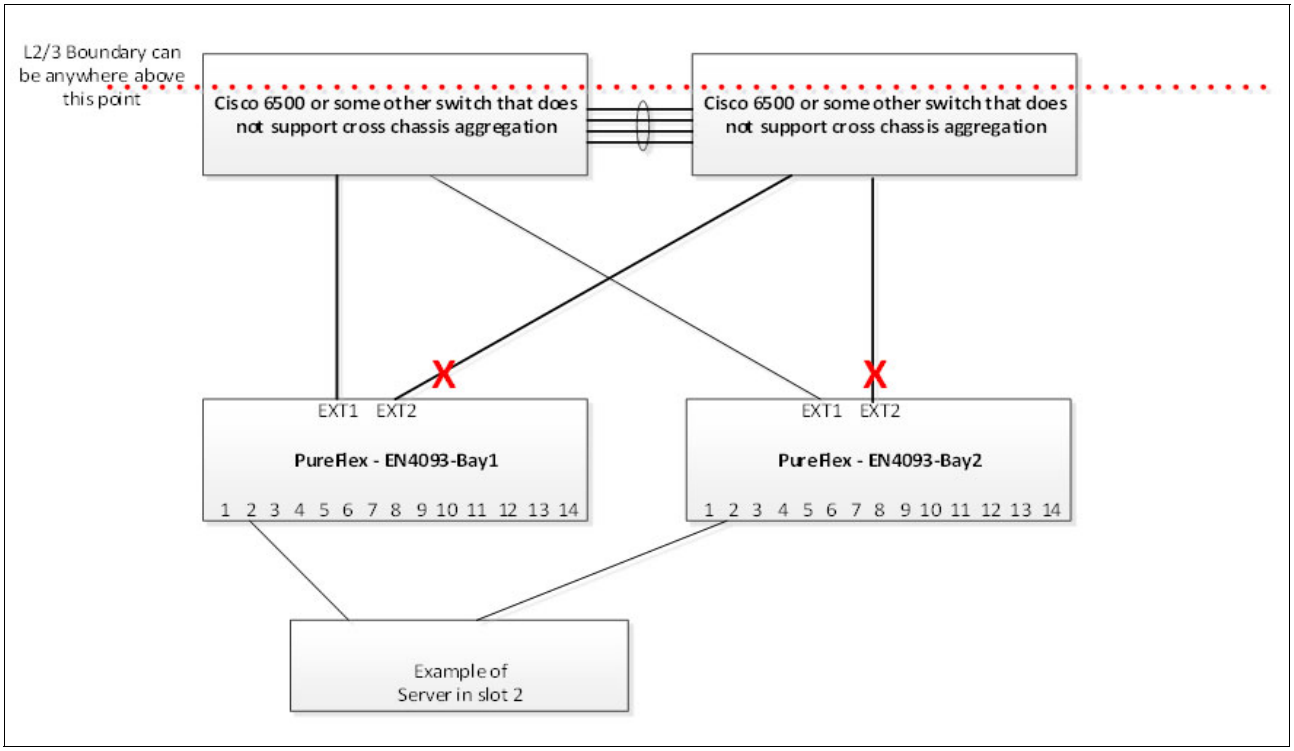


Figure 2-9 Network topology with STP and blocked uplinks

Figure 2-10 shows an excerpt of a configuration that implements this design. Unless disabled (or unless the number of VLANs exceeds the maximum supported number of STP groups) the STP configuration commands are generated automatically when the VLANs are created. When the VLAN number is greater than the maximum number of instances (for example, VLAN 2000), an arbitrary STP instance number is chosen and this number is displayed in an informational message.

```
interface port INT2,EXT1,EXT2
switchport mode trunk
switchport trunk allowed vlan 10,20
spanning tree stp 10 vlan 10
spanning tree stp 20 vlan 20
```

Figure 2-10 Spanning tree configuration commands

In this configuration, vLAG can still be used on the internal ports (such as INT2), and then aggregation-based active/active NIC teaming or bonding on the server can be used. Traditionally, this configuration includes configuring the server to use switch independent mode active/active NIC teaming (for example, Linux mode 5), or active/standby (such as Linux *bonding mode=1*) teaming. For more information about server NIC teaming options, see Chapter 9, “Integrating with hosts” on page 193.

2.2 Other design considerations

This section describes some preferred practices for designs that are not described in 2.1, “Sample topologies” on page 6.

This list is not meant to be an exhaustive list of all possible designs, and other configurations or designs are possible.

2.2.1 FCoE with vLAG

This section describes preferred practices for enabling FCoE within a vLAG environment, which includes the following design considerations:

- Flex Switch CN4093 switch
- Flex Switch EN4093 to G8264CS switches
- Flex Switch EN4093 to Nexus 5548 switches

Flex Switch CN4093 switch

The Flex Switch CN4093 switch provides connectivity to a Standard Ethernet Network and a Fibre Channel environment as an FCoE Gateway.

Figure 2-11 shows a SAN A and SAN B for Fibre Channel Multipath isolation. SAN A carries FCoE and Fibre Channel Traffic on the path vHBA-1 → CN4093-1 → SAN Fabric A and SAN B carries FCoE and Fibre Channel Traffic on the path vHBA-2 → CN4093-2 → SAN Fabric B. The vLAG Inter Switch Link (ISL) on the CN4093 Switches carries normal Ethernet traffic only and it is required to prune FCoE VLANs.

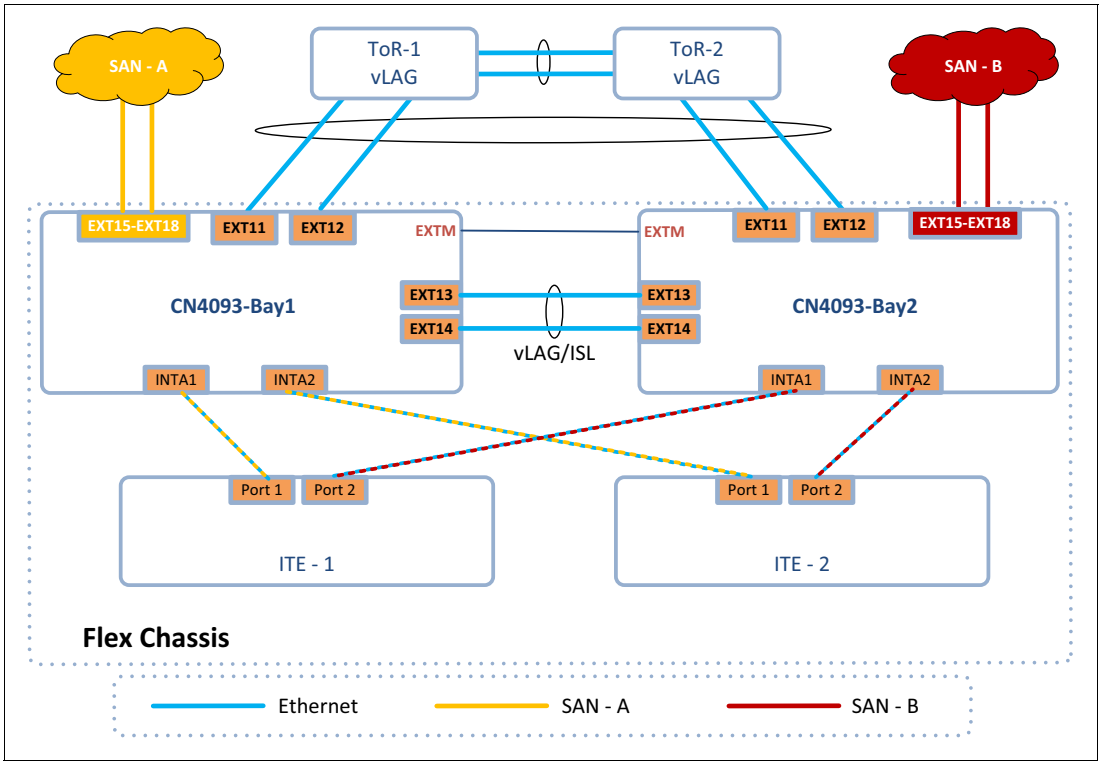


Figure 2-11 FCoE with vLAG that use CN4093 switches

Consider the following preferred practice guidelines:

- Ensure that vLAG ISL ports contain a non-production VLAN as native VLAN or PVID; for example, VLAN 4090 or VLAN 4094. Disable Spanning Tree for this VLAN to prevent it from participating in Spanning Tree. Do not add this VLAN to any other ports except for the vLAG ISL ports.
- Internal-node-facing ports (for example, INTA ports) on each of the EN4093 switches must have VLAN 1 as the native VLAN or PVID VLAN. It is used in the FCoE discovery process.
- Internal-node-facing ports (for example, INTA ports) on each of the CN4093 switches must contain the FCoE VLAN. It is commonly VLAN 1001 (SAN A) and VLAN 1002 (SAN B), but this configuration is not mandatory.
- If Spanning Tree is enabled on the CN4093 switch, disable Spanning Tree for the FCoE VLAN. The FCoE discovery VLAN 1 can have Spanning Tree enabled if it is needed.
- vLAG ISL ports must *not* include either of the two configured FCoE VLANs to prevent Fibre Channel fabric merging from occurring.
- Use loop guard across the vLAG ISL for loop detection and prevention.
- If Spanning Tree is enabled, use BPDU guard on the ITE (server) facing ports.

Example 2-5 shows an example configuration script for a CN4093 switch with vLAG and FCoE in NPV mode.

Example 2-5 Example configuration script for a CN4093 switch with vLAG and FCoE in NPV mode

```
hostname "CN4093-CH1-SW1”
system port EXT15-EXT18 type fc
!
cee enable
fcoe fips enable
!
interface port INTA1-INTA14,EXT15-EXT18
   switchport mode trunk
   switchport trunk allowed vlan 1,10,20,30,1001
   exit
!
vlan 1001
   name "FCoE VLAN"
   npv enable
   npv traffic-map external-interface EXT15-EXT18
   exit
!
interface port EXT1,EXT2
   description vLAG-ISL
   switchport mode trunk
   switchport trunk allowed vlan 10,20,30,4090
   switchport trunk native vlan 4090
   spanning-tree loopguard
   lacp key 4344
   lacp mode active
   exit
!
interface port EXT11,EXT12
   description Uplink-To-ToR1
   switchport mode trunk
   switchport trunk allowed vlan 10,20,30,999
```

```
   switchport trunk native vlan 999
   lacp key 5354
   lacp mode active
   exit
!
interface port INTA1-INTA14
   bpdu-guard
!
no spanning-tree stp 26 enable
spanning-tree stp 26 vlan 4090
!
no spanning-tree stp 112 enable
spanning-tree stp 112 vlan 1001
!
vlan 10
   name Data-Network-10
vlan 20
   name Data-Network-20
vlan 30
   name Data-Network-30
!
interface ip 127
   ip address 1.1.1.1
   enable
!
vlag enable
vlag tier-id 10
vlag hlthchk peer-ip 1.1.1.2
vlag isl adminkey 4344
vlag adminkey 5354 enable
```

Important: The **`no spanning-tree stp`** ***`xxx`*** commands can vary regarding the instance number. The **`show run | section vlan`** command displays which STP instance is associated to which VLAN to correctly identify the STP instance ID. In Example 2-5 on page 17, spanning-tree for the ISL VLAN (4090) and the FCoE VLAN (1001) is disabled.

Flex Switch EN4093 to G8264CS

The EN4093 allows for Ethernet and FCoE as a transit switch. The G8264CS switch can support Ethernet and Fibre Channel as an FCoE Gateway device.

Figure 2-12 shows a SAN A and a SAN B for Fibre Channel Multipath isolation. SAN A carries FCoE and Fibre Channel Traffic on the path vHBA-1 → EN4093-1 → G8264CS-1 → SAN A and SAN B carries FCoE and Fibre Channel Traffic on the path vHBA-2 → EN4093-2 → G8264CS-2 → SAN B. The vLAG Inter Switch Link (ISL) on the EN4093 and the G8264CS pairs of switches carries normal Ethernet traffic only and is required to prune FCoE VLANs.

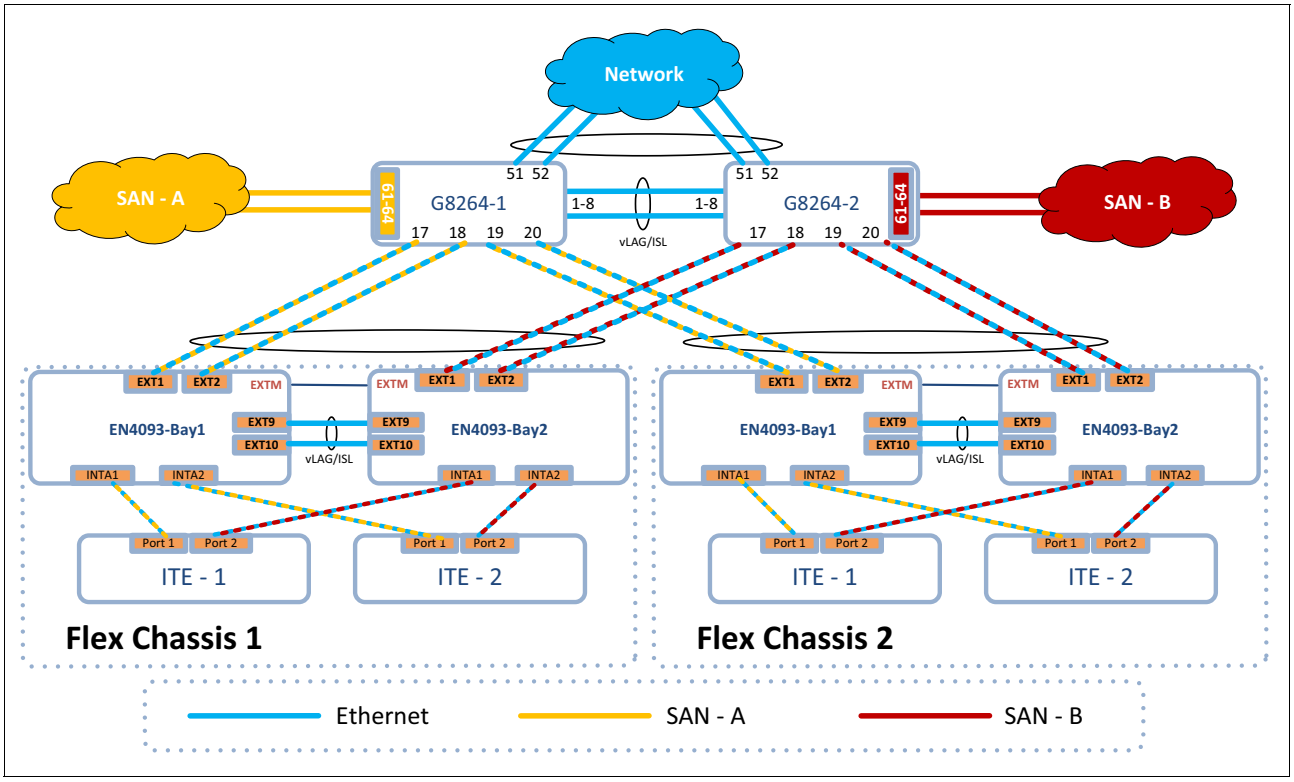


Figure 2-12 FCoE with vLAG that uses EN4093 and G8264 switches

Consider the following preferred practice guidelines:

- Ensure that vLAG ISL ports contain a non-production VLAN as the native VLAN and PVID; for example, VLAN 4090 or VLAN 4094. Disable Spanning Tree for this VLAN to prevent it from entering an STP block state or participating in Spanning Tree. Do not add this VLAN to any other ports other than the vLAG ISL ports.
- Internal-node-facing ports (for example, INTA ports) on each of the EN4093 switches must have VLAN 1 as the Native VLAN and PVID. It is used as the FCoE discovery VLAN.
- EN4093 external ports that face the network must contain VLAN 1, but can be trunked and tagged to carry the FCoE discovery information to the G8264CS switch for processing.
- Internal Node facing ports (for example, INTA ports) on each of the EN4093 switches must contain the FCoE VLAN. It is usually VLAN 1001 (SAN A) and VLAN 1002 (SAN B).
- If Spanning Tree is enabled on the EN4093 switch, G8264CS switch, or both, disable Spanning Tree for the FCoE VLAN. Optionally, if it is required to be enabled on the vLAG ISL ports, VLAN 1 can have Spanning Tree enabled if it is needed.
- vLAG ISL ports must not include either of the two configured FCoE VLANs to prevent Fibre Channel merging from occurring.
- Use loop guard across the vLAG ISL for loop detection and prevention.
- If Spanning Tree is enabled, apply bpdu-guard on the ITE (server) facing ports.

Example 2-6 shows an example configuration of an EN4093 switch in Easy Connect (EC) mode with vLAG and FCoE enabled.

Example 2-6 Example configuration of an EN4093 in EC Mode with vLAG and FCoE

```
hostname EN4093-Sw1
spanning-tree mode disable
cee enable
!
interface port ext9,ext10
   description vLAG-ISL
   switchport mode trunk
   switchport trunk allowed vlan 4090,4091
   switchport trunk native vlan 4090
   lacp key 5152
   lacp mode active
   exit
!
vlan 4090
   name Peer-Link
   exit
!
vlan 4091
   name EasyConnect
   exit
!
interface port inta1-inta14,ext1-ext2
   switchport access vlan 4091
   tagpvid-ingress
   exit
!
interface port ext1-ext2
   description Uplink-To-G8264CS-1
   lacp key 4344
   lacp mode active
   exit
!
interface ip 127
   ip address 1.1.1.1
   enable
   exit
!
vlag ena
vlag tier-id 10
vlag hlthchk peer-ip 1.1.1.2
vlag isl adminkey 5152
vlag adminkey 4344
enable
```

Example 2-7 shows a configuration of a G8264CS switch with vLAG plus FCoE and FC enabled.

Example 2-7 Example configuration of a G8264CS with vLAG plus FCoE and FC

```
hostname G8264CS-Sw1
!
interface port 17-20,61-64
   switchport mode trunk
   switchport trunk allowed vlan 1,1001
   exit
!
cee enable
fcoe fips enable
!
system port 61-64 type fc
!
interface port 17,18
   description To-Ch1-Sw1
   lacp key 1718
   lacp mode active
   exit
!
interface port 19,20
   description To-Ch2-Sw1
   lacp key 1920
   lacp mode active
   exit
!
vlan 1001
   name "FCoE VLAN"
   npv enable
   npv traffic-map external-interface 61-64
   exit
!
interface port 1-8
   description vLAG-ISL
   switchport mode trunk
   switchport trunk allowed vlan 10,20,30,4090
   switchport trunk native vlan 4090
   spanning-tree loopguard
   lacp key 1080
   lacp mode active
   exit
!
interface port 51,52
   description To-Core-Network
   switchport mode trunk
   switchport trunk allowed vlan 10,20,30,999
   switchport trunk native vlan 999
   lacp key 5152
   lacp mode active
   exit
!
interface port 17-20
   switchport mode trunk
```

```
   switchport trunk allowed vlan add 10,20,30
   bpdu-guard
   exit
!
no spanning-tree stp 26 enable
no spanning-tree stp 112 enable
!
vlan 10
   name Data-Network-10
vlan 20
   name Data-Network-20
vlan 30
   name Data-Network-30
!
interface ip 128
   ip address 1.1.1.1
   enable
!
vlag enable
vlag tier-id 1
vlag hlthchk peer-ip 1.1.1.2
vlag isl adminkey 1080
vlag adminkey 5152 enable
vlag adminkey 1718 enable
vlag adminkey 1920 enable
```

Important: The **`no spanning-tree stp 26 enable`** command correlates to VLAN 4090, which is used as the ISL VLAG Native VLAN ID.

The **`no spanning-tree stp 112 enable`** command correlates to VLAN 1001, which is used as the FCoE VLAN ID.

However, the **`show run | section vlan`** command displays the `stp` instance that is associated to which VLAN to correctly identify the `stp` instance ID.

Flex Switch EN4093 to a Nexus 5548 switch

The EN4093 switch allows for Ethernet and FCoE as a transit switch. The Cisco Nexus 5548 switch can support Ethernet and Fibre Channel as an FCoE Gateway device.

Figure 2-13 shows a SAN A and SAN B for Fibre Channel Multipath isolation. SAN A carries FCoE and Fibre Channel Traffic on the path vHBA-1 → EN4093-1 → Nexus 5548-1 → SAN A. SAN B carries FCoE and Fibre Channel Traffic on the path vHBA-2 → EN4093-2 → Nexus 5548-2 → SAN B. The vLAG Inter Switch Link (ISL) on the EN4093 switch and the vPC ISL on the Nexus 5548 pairs of switches carry only normal Ethernet traffic and are required to prune any FCoE VLANs.

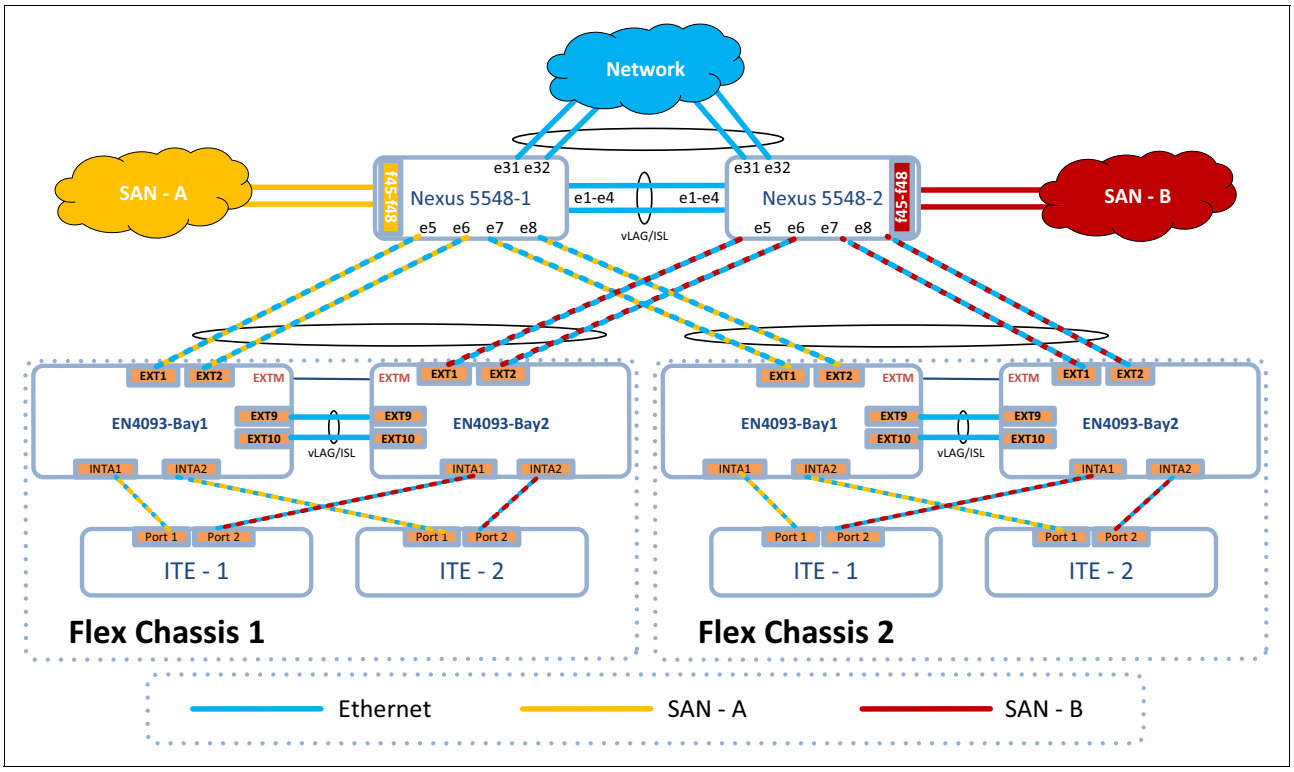


Figure 2-13 FCoE with vLAG that uses EN4093 and Nexus switches

Consider the following preferred practice guidelines:

- For more information about Nexus Peer-Link and vPC preferred practices, see the Cisco documentation. An example of a vPC and PortChannel is provided in Example 2-8 on page 24 and Example 2-9 on page 25 as an option when you are connecting to a Flex Chassis pair of EN4093 switches that are enabled with vLAG and FCoE.
- Ensure that vLAG ISL ports should contain a non-production VLAN as the native VLAN and PVID; for example, VLAN 4090 or VLAN 4094. This VLAN should also have Spanning Tree disabled to prevent it from ever entering an STP block state or participating in Spanning Tree. This VLAN should not be added to any other ports other than the vLAG ISL ports.
- Internal Node-facing ports (for example, INTA ports) on each of the EN4093s must have VLAN 1 as the Native VLAN / PVID, which is used as the FCoE discovery VLAN.
- EN4093 external ports that are facing the Network must contain VLAN 1 but can be trunked or tagged to carry the FCoE discovery information to the Nexus 5548 Switches for processing.
- Internal Node facing ports (for example, INTA ports) on each of the EN4093s must contain the FCoE VLAN; usually VLAN 1001 (SAN A) and VLAN 1002 (SAN B).

- If Spanning Tree is enabled on the EN4093 the FCoE VLAN should have Spanning Tree disabled. Optionally, VLAN 1 (if required to be enabled on the vLAG ISL ports) can have Spanning Tree enabled if needed or is required.
- vLAG ISL ports do not include either of the two configured FCoE VLANs to prevent Fibre Channel merging from occurring.
- Loop Guard across the vLAG ISL for loop detection and prevention.
- If Spanning Tree is enabled, applying `bpdu-guard` on the ITE facing ports is recommended.
- The Nexus 5548 interfaces that are facing the EN4093 Switches must have the **`priority-flow-control mode on`** command used to interoperate with the EN4093 Switches when FCoE is enabled over those interfaces (a lack of this command can result in PFC not being properly negotiated and severe FCoE performance issues to occur).

Example 2-8 shows an EN4093 with vLAG and FCoE enabled.

Example 2-8 Example configuration of an EN4093 with vLAG and FCoE

```
hostname EN4093-Sw1
spanning-tree mode disable
cee enable
!
interface port ext9,ext10
   description vLAG-ISL
   switchport mode trunk
   switchport trunk allowed vlan 4090,4091
   switchport trunk native vlan 4090
   lacp key 5152
   lacp mode active
   exit
!
vlan 4090
   name Peer-Link
   exit
!
vlan 4091
   name EasyConnect
   exit
!
interface port inta1-inta14,ext1-ext2
   switchport access vlan 4091
   tagpvid-ingress
   exit
!
interface port ext1-ext2
   description Uplink-to-Nexus-1
   lacp key 4344
   lacp mode active
   exit
!
interface ip 127
   ip address 1.1.1.1
   enable
   exit
!
vlag ena
```

```
vlag tier-id 10
vlag hlthchk peer-ip 1.1.1.2
vlag isl adminkey 5152
vlag adminkey 4344 enable
```

Example 2-9 shows a Nexus 5548 interface/PortChannel with vPC and FCoE enabled.

Example 2-9 shows a configuration of a Nexus 5548 interface/PortChannel with vPC and FCoE

```
interface Ethernet1/5
   description PureFlex-Ch1-Sw1-Port-EXT1
   switchport mode trunk
   switchport trunk allowed vlan 1,10,20,30,1001
   channel-group 5 mode active
   priority-flow-control mode on
!
interface Ethernet1/6
   description PureFlex-Ch-1-Sw1-Port-EXT2
   switchport mode trunk
   switchport trunk allowed vlan 1,10,20,30,1001
   channel-group 5 mode active
   priority-flow-control mode on
!
interface port-channel5
   description PF-CH-1-Sw1-Ports-EXT1&EXT2
   switchport mode trunk
   switchport trunk allowed vlan 1,10,20,30,1001
   spanning-tree port type edge trunk
   priority-flow-control mode on
   speed 10000
   vpc 5
```

2.2.2 Isolated management network

The use of a separate management network is always a preferred practice for isolation of data and management environments. An isolated management network can be used in a lights-out environment to provide out-of-band connectivity to locate and troubleshoot issues that might span over the data network and compute nodes. In today's data centers, this network often consists of a management VLAN that uses 1 Gb connectivity.

Additionally, when Flex chassis are used in the network, it is critical that the CMM, IMM, and the management interfaces for switches in IO bays 1 - 4 are not placed in a data VLAN or subnet that is used by the hosts in the Flex System chassis. The CMM uses IP forwarding and proxy ARP functionality to provide network access to internal chassis management interfaces.

If operating systems (OSs) are on the same network, the CMM can respond to ARPs by the OS, which leads to the OS losing connectivity to the rest of the network. Because of the potential loss of OS connectivity, a configuration that places OSes on the same VLAN/subnet as a CMM, IMM, and switch management interface is not supported.

Figure 2-14 shows Flex Chassis Management Components and RackSwitch components that are connecting to a separate 1 Gb Management for out-of-band management.

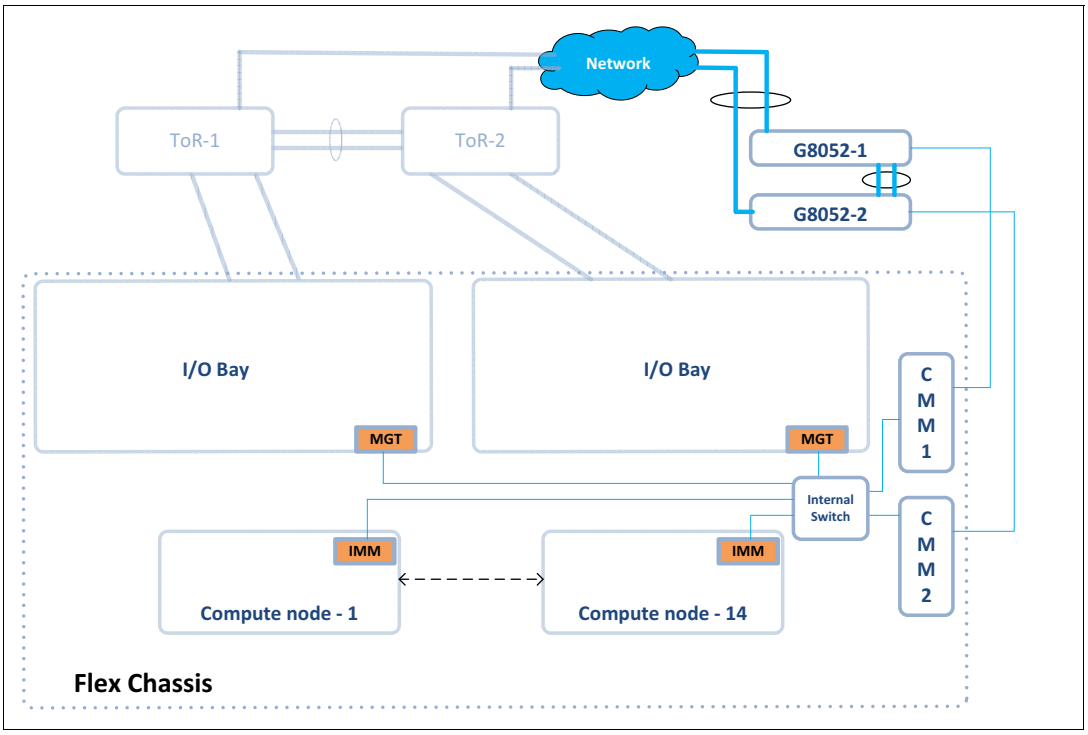


Figure 2-14 Out of Band 1G Management connectivity

When you use a separate management environment for out-of-band connectivity, it is important to remember that the management network still can consist of high availability features, such as vLAG. Consider the following points about the management network:

- If an HA environment is used in the management network between a pair of Rack Switches, vLAG and Spanning Tree can be enabled.
- Spanning Tree `portfast` and `bpdu-guard` should be enabled on all access devices, such as Server or Storage NICs and Management components (for example, Flex Chassis CMM).
- Loop Guard across the vLAG ISL for loop detection and prevention, if used.
- If a pair of management switches is used that do not have a dedicated management port for use as the vLAG health check (for example, the RackSwitch G8052), a separate physical link should be configured to carry the health check heartbeat between a pair of vLAGed switches.

Example 2-10 shows the configuration of a G80 52 1 G Management Switch.

Example 2-10 Example configuration of a G8052 1G management switch

```
hostname G8052-MGMT-Sw1
!
interface port 51,52
   description vLAG-ISL
   switchport mode trunk
   switchport trunk allowed vlan 1021,4090
   switchport trunk native vlan 4090
   spanning-tree guard loop
```

```
   lacp key 5152
   lacp mode active
   exit
!
vlan 4090
   name vLAG-ISL
   exit
!
vlan 999
   name Native_VLAN
   exit
!
vlan 1021
   name Management_VLAN
   exit
!
vlan 1022
   name vLAG-HC_VLAN
   exit
!
no spanning-tree stp 6 enable
spanning-tree stp 6 vlan 1022
!
spanning-tree stp 26 vlan 4090
no spanning-tree stp 26 enable
!
interface port 49,50
   description Uplink-To-Network
   switchport mode trunk
   switchport trunk allowed vlan 1021,999
   switchport trunk native vlan 999
   lacp key 4950
   lacp mode active
   exit
!
interface port 1-47
   switchport access vlan 1021
   spanning-tree portfast
   bpdu-guard
   exit
!
interface port 48
   description Health-check-link
   switchport access vlan 1022
   spanning-tree portfast
   exit
!
interface ip 1
   ip address 1.1.1.1 255.255.255.0
   vlan 1022
   enable
   exit
!
vlag ena
vlag hlthchk peer-ip 1.1.1.2
```

```
vlag tier-id 100
vlag isl adminkey 5152
vlag adminkey 4950 enable
```

Important: The **no spanning-tree stp 26 enable** command correlates to VLAN 4090, which is used as the VLAG ISL Native VLAN ID. However, the **show run | section vlan** command displays the stp instance that is associated to which VLAN to correctly identify the stp instance ID.

2.2.3 vLAG versus Stacking

Virtual Link Aggregation (vLAG) and Switch Stacking have there own unique attributes that have advantages that are specific to individual customer requirements. Consider the following points about each option:

- vLAG supports most of the same available features as a stand-alone switch, whereas Stacking tends to support a smaller subset of those features. For more information about unsupported features, see the Application Guide that is specific to the required switch type.
- Because vLAG Switches are managed as independent devices, upgrades can be performed independently, which creates a staggered upgrade approach. However, when the Master Stack Switch is upgraded, Stacked Switches upgrade all members of the Stack and reload the entire stack, which causes a potential outage to all hosts behind the stack. There are ways around this issue by creating a dual stack where some switches are in stack 1 and others are in stack 2. However, this configuration does not allow for east-west traffic between stacks to be allowed unless there is a connection between the stacks. This configuration must be carefully planned as to not create a loop topology after it is connected to the upstream network.
- Local preference for Stacking is not supported on any of the embedded Flex Chassis Switches or the Top of Rack Switches. However, vLAG supports local preference, which can have significant advantages in efficient link utilization. For more information about local preference, see 4.1, “Virtual Link Aggregation Group considerations” on page 48.

2.2.4 Easy Connect

Easy Connect is a simple configuration mode that is implemented on Lenovo Networking Ethernet and Converged switches that enables easy integration of Flex System integrated networking with Cisco, Juniper, and other vendor data center networks. Easy Connect makes connecting to core networks simple while enabling advanced in-system connectivity at the network edge. It also allows administrators to allocate bandwidth and optimize performance.

Easy Connect types and uses cases

With Easy Connect enabled, the switch becomes a simple I/O module that connects servers and storage with the core network. It aggregates compute node ports and behaves similarly to Cisco Fabric Extension (FEX) by appearing as a “dumb” device to the upstream network. With Easy Connect enabled, the upstream network and the attaching hosts are responsible for managing all VLAN assignments and tagging. This loop-free connectivity requires no extra configuration and helps provide economical bandwidth use with prioritized pipes and network virtualization.

The following types of Easy Connect are available:

- `tagpvid-ingress`: This type is one of the most common and robust forms of Easy Connect because it allows for vLAG.

  Figure 2-15 shows an example of how to implement `tagpvid-ingress` Easy Connect with vLAG on a pair of EN4093s and G8264 Top of Rack Switches (Nexus 5k Switches and others can also be used when the ToR Switches appear as a single entity; that is, Stack, vLAG, VSS, vPC, MCLAG, and so on).

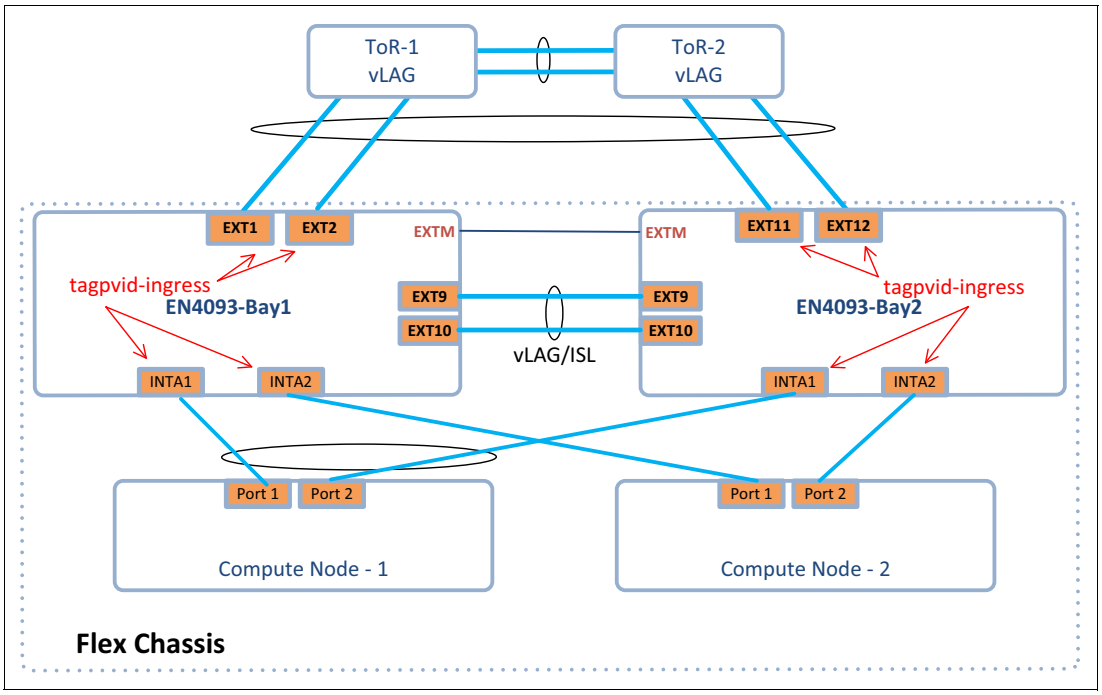


Figure 2-15 Easy Connect with tagpvid-ingress and vLAG

Example 2-11 on page 30 shows how to successfully deploy a `tagpvid-ingress` with vLAG. To successfully deploy this configuration, the upstream network components should also be running some type of Virtualization, such as Stacking, vLAG, vPC, or MC-LAG, depending on the platform of choice.

The following preferred practices are guidelines for implementing `tagpvid-ingress` with or without vLAG:

- All of the normal rules for the VLAG ISL apply.
- A non-production VLAN should be used as the `tagpvid-ingress` Native VLAN or PVID; for example, VLAN 4091 or VLAN 4092. This VLAN also should have Spanning Tree disabled if it is within a Flex Chassis and enabled if it is within Top of Rack Switches to use some of the Spanning Tree protection, such as loop guard across the vLAG ISL and BPDU Guard on all Server and Flex Chassis facing ports.
- As shown in Figure 2-15, you can be seen that regardless of whether you are running `tagpvid-ingress` or Layer 2 standard switching, both options support similar topologies.

Example 2-11 Example configuration of an EN4093 with tagpvid-ingress and vLAG

```
hostname EN4093-Sw1
spanning-tree mode disable
interface port ext9,ext10
   switchport mode trunk
   switchport trunk allowed vlan 4090,4091
   switchport trunk native vlan 4090
   lacp key 5152
   lacp mode active
vlan 4090
   name ISL-Peer-Link
vlan 4091
   name EasyConnect
interface port inta1-inta14,ext1-ext2
   switchport access vlan 4091
   tagpvid-ingress
interface port ext1-ext2
   lacp key 4344
   lacp mode active
interface ip 127
   ip address 1.1.1.1
   enable
vlag ena
vlag tier-id 10
vlag hlthchk peer-ip 1.1.1.2
vlag isl adminkey 5152
vlag adminkey 4344 enable
```

- SPAR: Switch Partitioning includes two operating modes: Pass-through Domain (which uses Q-n-Q and is a form of Easy Connect) and Local Domain, which uses traditional VLANs or sometimes referred to as a Customer VLAN (cVLAN).
- Unified Fabric Port (UFP (Tunnel Mode)): This type provides for the ability to carve up a 10 G Port into four Virtual Lanes, which are also known as vNICs. UFP Tunnel Mode is also a form of Q-in-Q and is another mode of Easy Connect.
- vNIC™ Virtual Fabric Mode: This mode is more of an older feature to UFP that supports bidirectional bandwidth metering and bandwidth control of the individual vNICs on the Host to be manipulated at the switch port. Before UFP was supported, vNIC VF Mode was the vNIC of choice by many customers, which also runs Q-n-Q (Easy Connect).
- The vNIC Groups can also be used to support physical ports, which allows for grouping INT and EXT Ports in this Easy Connect solution.

For more information about the types of Easy Connect, see *NIC Virtualization in Flex System Fabric Solutions*, SG24-8223, which is available at this website:

http://lenovopress.com/sg248223

Easy Connect versus full L2 switching

Although Easy Connect is a simple and recommended approach to keeping simplicity and ease-of-use, it might not always meet all requirements. Full Layer 2 (L2) Switching that includes managing of VLANs, Spanning Tree, and possibly Layer 3 has more capability and allows for easier troubleshooting over that of Q-n-Q.

Consider the following advantages of each option:

- Layer 2 Switch Mode:
  - Allows for ongoing VLAN management and manipulation of Native VLAN/PVID ID on a per port basis (inner VLAN aware).
  - Allows for FCoE FIPs detection for security
  - Allows for Spanning Tree Management (option to disable)
- Easy Connect Mode
  - One time configure or Plug and Play (SI4093) options.
  - Allows for VLAN Independent for simple management (is not aware of inner tag).
  - Allows for Spanning Tree to be disabled.
  - Allows for vLAG in some methods of Easy Connect.
  - FCoE CEE enabled for ETS and PFC exchange only; FCoE FIPs not possible because it requires inner VLAN ID.
- Combination of L2 and Easy Connect Mode:
  - Allows for L2 and Easy Connect to be configured on the same Switch (often seen in UFP Mode or when looking for separation of Ethernet and FCoE).
  - Easy Connect and standard L2 networking require independent physical ports because Easy Connect converts a physical port into a Q-n-Q port. After the command **`tagpvid-ingress`** was run on that interface, it removes the ability to allow older VLANs to be used on the same port.

3

Layer 1 technologies

This chapter provides information about the physical connections that are available on Lenovo Networking switches. It includes information about cables, transceivers, and low-level port settings for these connections, and specific considerations and guidelines.

This chapter includes the following topics:

3.1 Considerations for cabling and transceivers

This section describes the physical cabling requirements and other considerations for the various ports on Lenovo Networking switches.

3.1.1 10/100/1000 Mb and 1 Gb-only ports

Ports in the 10/100/1000 Mb - 1 Gb speed range are available in two types: copper RJ45 (both built-in and SFP-based) and SFP-based fiber. All Lenovo Networking switches have one or more built-in RJ45 ports. For some switches, such as the G8264 switch, this port is a single 10/100/1000 Mb dedicated management port, and, for others, such as the G8052 switch, these ports are 48 RJ45 10/100/1000 Mb data ports. Many Lenovo switches also have SFP/SFP+ ports that support installing various 1 Gb SFP modules (fiber and RJ45 copper). This section describes the characteristics of these ports and modules.

Dedicated RJ45 Management ports

For switches that have a dedicated RJ45 Ethernet management port, this port supports 10/100/1000 Mb speeds and is for switch management *only*. It has no connection to the data ports; that is, packets that are coming in and out of the dedicated Ethernet management port can *never* be switched into or out of any of the data ports, and vice versa.

Exception: The exception to this packet flow between management and data ports is the management port (EXT11) on the Virtual Fabric 10Gb Switch Module for BladeCenter. Although designed to be primarily used for management, that management port is connected into the switch fabric and packets can pass between it and the other data ports on the switch if common VLANs are used.

The dedicated management ports do not support any form of tagging, send *only* untagged packets, and expect to receive untagged packets. When you are looking at the VLANs that are allowed on ports on a switch (for example, by using the **`show int info`** or **`show int trunk`** commands), dedicated management ports show that they are associated with VLAN 4095. For all externally visible dedicated management ports, consider this VLAN to be a place holder to show that the port is not connected to the fabric; that is, it is not really sending and receiving packets on VLAN 4095.

For internal dedicated management ports on BladeCenter, VLAN 4095 tagged packets are used between the switch and the Management Module in the BladeCenter. However, packets that are tagged with VLAN 4095 never leave the BladeCenter (the uplink of the Management module also sends and receives untagged traffic *only*).

Flex System embedded switches: For Flex System embedded switches that have a dedicated internal management port toward the CMM (MGT) and a dedicated RJ45 management uplink (EXTM), both ports appear as being on VLAN 4095 when you are looking at certain command output, such as the output of a **`show int trunk`** command. Although this output might make it look like these two ports are on a common VLAN and might potentially pass traffic, they are isolated from each other. No traffic ever passes between these two out-of-band management ports.

Built-in RJ45 data ports

RJ45 data ports adhere to standard IEEE 802.3 10/100/1000 Mb rules and default to auto negotiate. These ports are normal user data ports that can be configured and used by end nodes to pass packets between ports. Top of Rack switches with built-in RJ45 data ports include the RackSwitch G8000, G8052, G7028, and G7052. For enclosure-based products, these switches include the Flex System EN2092 1Gb Ethernet Scalable Switch and the BladeCenter Layer 2/3 Ethernet Switch Module, BladeCenter Layer 2-7 Gigabit Ethernet Switch Module, BladeCenter 1/10Gb Uplink Ethernet Switch Module, Cisco CIGESM, and the Cisco 3*XXX* family.

SFP RJ45 and optical modules

All 1 Gb SFP ports that use approved SFP modules (RJ45 or fiber) support 1 Gbps speed only. No support for 10/100 is available in any SFP/SFP+ ports on Lenovo switches. Lenovo switches support only Lenovo approved SFP modules, and usually do not support other vendors' SFP modules. Attempting to use non-approved SFP modules usually causes that port to become unusable until the non-supported SFP module is removed and replaced with an approved SFP. All Lenovo SFP fiber modules use an LC type of connector. The type of fiber cabling that is supported by each SFP module varies by the module type. For more information about types of fibers and distances that are supported for various modules, see Table 3-1 on page 39.

Internal 1 Gb ports on enclosure-based switches

The enclosure 1 Gb switches Flex System EN2092 and the BladeCenter L2/3 ESM, L2-7 ESM, 1/10Gb Uplink ESM, Cisco CIGESM, and the Cisco 3*XXX* family also have internal facing ports that do not have any physical RJ45 connection. These ports are hard-wired via the mid-plane of the enclosure between the enclosure-based hosts and the enclosure-based switches. These ports default to 1 Gbps, and often can hardcoded to lower speeds, but this change is rarely done.

For SFP optical modules, all connectors use a physical LC format connector. The type of physical connector on the other side of the link does not need to be an LC connector. It can be SC or some other format if the other side of the link has the same optical type (that is, SX or LX). Attempting to attach unlike modules (that is, connecting an SX to an LX transceiver) causes a poor connection or no connection.

3.1.2 10 Gb connections

For almost all 10 Gb switches, the 10 Gb ports use SFP+ modules, except for the following switches:

- G8264T; uses built-in 10 GbaseT RJ45 connectors
- Cisco 3110X for BladeCenter; uses X2 based modules
- 6-port 10 Gb ESM for BladeCenter; uses XFP-based modules
- Internal 10 Gb ports on embedded switch modules. As with embedded internal 1 Gb facing ports, these connections are hard-wired between the switch module and the enclosure-based server via the mid-plane of the enclosure.

The 10 Gb switches that support SFP+ based ports support a range of Lenovo transceivers, in addition to some Direct Attach Cable (DAC), which is also known as TwinAx). This section describes these transceivers and cables with some considerations on usage.

Optical 10 Gb

The 10 Gb switches support a range of 10 Gb optical modules, and support might vary between switch models. As with 1 Gb SFP modules, Lenovo switches support only Lenovo approved SFP+ modules, and attempting to use a non Lenovo approved SFP+ module usually renders that port inoperable until an approved SFP+ module is installed. All Lenovo SFP+ modules use an LC type of connector. The type of fiber cabling that is used by each module varies by the module type. For more information about the types of fibers and distances that are supported, see Table 3-1 on page 39.

DAC

Unlike SFP+ modules, Lenovo does not lock out other vendors DACs, but does have specific limitations on types (active or passive) and lengths of DACs supported. Typically, 5 meter or shorter passive DACs work on all Lenovo switches that have SFP+ ports. Some Lenovo switches also support longer DAC cables, in addition to Active DACs. For more information about what your switch supports, see the installation guide and other documentation for that product.

In most instances, any MSA-compliant DAC works in Lenovo switches.

Because DACs are a single cable that potentially connects two different model or vendor switches, any DAC used must be compatible with *both* sides of the connection. For example, if one side supports a 10 meter DAC and the other side supports only a maximum of a 5 meter DAC, you cannot use a 10 meter DAC (you must use a 5 meter or shorter DAC). The same condition is true of active and passive DAC. If one side supports active and passive and the other side supports only passive, you can use only a passive DAC. In the rare instance where the two sides do not have a common supported DAC (for example, one side supports *only* active DAC, and the other side supports *only* passive DAC), DAC is not suitable for this environment. Instead, consider an optical SFP+ module.

10GBaseT

Only the G8264T switch supports 10GBaseT ports. These RJ45 10GBaseT ports can be 1 Gb or 10 Gb and have a default setting for auto negotiation. The 1 Gb connections can use Cat5 or better cabling for up to 100 meter distances. Use Cat6 or better cable for 10 Gb connections. Cat6 cable supports up to 50 m at 10 Gb speeds, and Cat6A cable supports up to 100 m at 10 Gb.

10 Gb ports

Consider the following points regarding 10 Gb ports:

- At the time of this writing, 10GbaseT transceivers are not supported in any Lenovo switches. This restriction is primarily a power limitation. If you require 10GbaseT ports, one option is to use the G8264T model switch that has 48 built-in 10GbaseT ports.

  The 10GBaseT ports on the G8264 are 100 Mb, 1 Gb, and 10 Gb only and do not support 10 Mb.

- Although most 10G SFP+ ports also support 1 Gb SFP modules, some ports on specific devices *do not*. The following devices do not support 1 Gb SFP modules in their SFP+ ports:
  - The three SFP+ ports on the 1/10 Uplink Ethernet Switch Module for BladeCenter.
  - The SFP+ ports on a dual port 10 Gb daughter card on a G8000.
  - Omni ports on the converged switches (currently ports EXT11-EXT22 on the Flex System CN4093 switch, and ports 53-64 on the RackSwitch G8264CS switch).
- 10/100 speeds are not supported for SFP+ ports on Lenovo switches.

- All Lenovo SFP+ 10 Gb modules use LC connectors. The other side of the link can be any type of connector (for example, LC, SC) if the optical type is the same (for example SR signaling on both sides, or LR signaling on both sides). Attempting to attach unlike modules (that is, connecting an SR transceiver to an LR transceiver) results in a poor connection or no connection at all.

3.1.3 40 Gb connections

Lenovo uses a QSFP+ format for all of its 40 Gb connections and supports DAC and optical modules for these ports.

Optical 40 Gb

As with 1 Gb and 10 Gb transceiver modules, Lenovo switches support only Lenovo approved QSFP+ modules. Attempting to use a non Lenovo approved QSFP+ module often renders that port inoperable until an approved QSFP+ modules is installed. The physical connection on xSR4-based (SR4, iSR4, and eSR4) optical 40 Gb modules is MPO (which is also referred to by the trademarked name of MTP). The LR4 optical module uses an LC connector.

DAC

The following rules for 40 Gb DAC (which is also known as TwinAx) are similar to 10 Gb DAC:

- Lenovo does not lock out other vendors QSFP+ based DACs.
- Lenovo does have restrictions on type (Active/Passive) and length.
- Both devices on each side of a QSFP+ DAC cable must agree on supported type (Active/Passive) and maximum length.

Most Lenovo QSFP+ 40 Gb ports also support being changed to a 4 x 10 Gb mode of operation. In that mode, they support a 4 x 10 Gb break out DAC (1 x QSFP+ male connector on one end, and 4 x SFP+ male connectors on the other end).

40 Gb ports

Consider the following points regarding 40 Gb ports:

- Most Lenovo 40 Gb ports support being operated as 1x 40 Gb port, or as 4x 10 Gb ports. To change between 1x 40 Gb and 4x10 Gb mode, use the **`boot qsfp-40Gports X`** or **`no boot qsfp-40Gports X`** commands (X = the port number). How the 40 Gb port is configured for speed can be seen with the **`show boot qsfp`** command, and any time the mode is changed, a reload is necessary for the change between modes can take effect.
- Use the command **`show boot qsfp`** to see the current mode for each QSFP+ port and the mode it assumes after the next reload.
- One exception to the ability of a Lenovo QSFP+ port to be set to a 4x 10 Gb port is the G8332 switch. For this switch, the first port and last 7ports (port 1 and ports 26 - 32) support 40 Gb only and cannot be set to 4x 10 Gb mode. Inversely, ports 2 - 25 can be set for 4x 10 Gb mode on the G8332 if wanted (but default to 40 Gb).
- For all Lenovo switches that support 40 Gb ports (except the G8332), the port numbering has gaps to support being separated into 4 x 10 Gb ports; for example, for a G8264 where the 40 Gb ports are the first physical ports on the switch, the ports default to being numbered as 1, 5, 9, and 13, and then 17 - 64 (where 1, 5, 9, and 13 represent the 40 Gb QSFP+ ports, and 17 - 64 are the 48 10 Gb SFP+ ports).

If any of the 40 Gb ports are converted to 4 x 10 Gb ports, the numbers are expanded out to represent that conversion. For example, if ports 1 and 5 were converted to 4x 10 Gb ports, the numbering is 1, 2, 3, 4, 5, 6, 7, 8, 9, 13, and then, 17 - 64, where 1 - 8 now represent the 4x 10 Gb ports that are part of the physical 40 Gb ports 1 and 5, 9 and 13 are still straight 40 Gb ports, and 17-64 remain 10 Gb SFP ports.

- For the G8332 switch, the port number formatting of the 40 Gb ports is different from the other 40 Gb ports on other Lenovo switches and has the following considerations:
  - By default, all 40 Gb ports are numbered 1 - 32.
  - When ports are placed into 4x 10 Gb mode and the switch is reloaded for the change to take effect, the numbering for these four converted 10 Gb ports is X/1 - X/4 (where X is the base port number, and the /1 through /4 are for physically broken out 10 Gb ports). For example, if port 2 were converted to 4x 10 Gb ports, the numbering becomes 2/1, 2/2, 2/3, and 2/4 after the reload to take effect.
  - Only ports 2 - 25 can be broken out in to 4x 10 Gb modes. Port 1 and ports 26 - 32 cannot be broken out to 4x 10 Gb; the ports can operate in 40 Gb mode only.
- When a 40 Gb port is used as 4x 10 Gb ports, it is common to use a break out fiber or DAC, as shown in Figure 3-1 and Figure 3-2

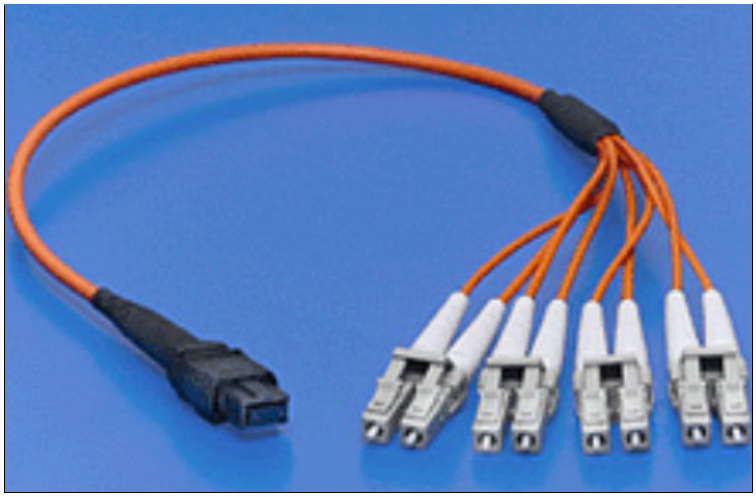


Figure 3-1 40 Gb *QSFP+ MPO connector to 4x 10 Gb LC connectors*

Figure 3-2 40 Gb *QSFP+ DAC to 4x 10 Gb SFP+ DAC connectors:*

- The 40 Gb optical modules are available in various types, such as SR4, eSR4, and iSR4. Attaching a like-to-like module can be done (that is, SR4 to SR4); however, mixing different module types on each end might lead to issues.

  If you stay within 100 m, attaching the different variations of SR4 works.

- When you are using an optical breakout cable and configuring a 40 Gb port for 4x 10 Gb ports, the use of one of the SR4 variations (SR4, eSR4, or iSR4) on the 40 Gb side and attaching to a 10 Gb SR optical module has the following limitations:
  - Attaching to any SFP+ based SR module works, but attaching to some older non-SFP+ 10 Gb SR format modules (for example, X2 or Xenpak) might not work.
  - When you are using an SR4 or an iSR4 and breaking this configuration out to 10 Gb SFP+ SR modules, it might be necessary to have some minimum length cable to attenuate the signal from the SR4 module to not over drive the 10 Gb SR module.
  - When you are using an eSR4 for this type of breakout, you often can attach to SR modules of any type (SFP+, X2, or Xenpak)

3.1.4 Transceiver considerations

This section describes transceiver considerations.

Supported transceiver modules and limitations

For more information about part numbers for 1 Gb, 10 Gb and 40 Gb transceivers for Lenovo Networking switches, see the documentation for that specific product. Lenovo locks out transceivers other than those from Lenovo, IBM, or BNT to ensure only fully tested and approved modules are used. For more information about determining supported modules for a switch, see this website:

http://public.dhe.ibm.com/common/ssi/ecm/lq/en/lqf12363usen/LQF12363USEN.PDF

Supported light levels on optical modules

When optical transceivers are used, it is important to understand what is acceptable for send and receive light levels. If a module is transmitting or receiving unexpected light levels, it can cause errors and in the case of excessive transmit light levels, premature failure of a module.

One of the more common issues with fiber connections is low light level, but how low is too low? Table 3-1 lists expected minimum and maximum light levels. If you receive signaling below and above these values, investigate and resolve the cause (most frequently dirty or defective optical cabling is the cause of low light, but it can also be caused by an incorrect cabling type or by bad SFP/SFP+/QSFP+ optical modules). Although these levels are part of the specifications, not all optics are equal and some might have issues when they are getting close to these levels. A good practice is to avoid running too close to the highs or lows that these numbers represent.

Table 3-1 General light levels and distances for common optical modules

Speed	Module P/N	TX Output Level	RX Input Level	Max Distances
1 Gb	SX SFP	112 uW (minimum) 501 uW (maximum)	20 uW (minimum) 1000 uW (maximum)	275 m OM1 MMF 550 m OM2 MMF
1 Gb	LX SFP	112 uW (minimum) 501 uW (maximum)	10 uW (minimum) 501 uW (maximum)	10k SMF
10 Gb	SR SFP+	186 uW (minimum) 794 uW (maximum)	102 uW (minimum) 794 uW (maximum)	26 m FDDI grade MMF 33 m OM1 MMF 82 m OM2 MMF 300 m OM3 MMF 400 m OM4 MMF
10 Gb	LR SFP+	151 uW (minimum) 1122 uW (maximum)	38 uW (minimum) 1112 uW (maximum)	10k SMF

Speed	Module P/N	TX Output Level	RX Input Level	Max Distances
10 Gb	ER SFP+	339 uW (minimum) 2512 uW (maximum)	26 uW (minimum) 794 uW (maximum)	40k SMF
40 Gb	SR4 QSFP+	174 uW (minimum) 1738 uW (maximum)	112 uW (minimum) 1738 uW (maximum)	100 m OM3 MMF 150 m OM4 MMF
40 Gb	iSR4 QSFP+	158 uW (minimum) 1738 uW (maximum)	1122 uW (minimum) 1738 uW (maximum)	100 m OM3 MMF 150 m OM4 MMF
40 Gb	eSR4 QSFP+	178 uW (minimum) 794 uW (maximum)	102 uW (minimum) 794 uW (maximum)	300 m OM3 MMF 400 m OM4 MMF
40 Gb	LR4 QSFP+	200 uW (minimum) 1698 uW (maximum)	43 uW (minimum) 1698 uW (maximum)	10k SMF

Important: Not all Lenovo switches support every module that is listed in Table 3-1 on page 39, and some switches have limitations on how many of a certain type of module can be installed at one time (for example, the G8264 supports up to a maximum of six ER modules to be installed at one time).

Also, some optics might exceed these ratings for minimums, maximums, or both. For more information about limitations and exact minimums and maximums for your modules and switches, see the documentation.

On Lenovo switches, current light levels for a specific optical module can be seen via a GUI and command-line interface (CLI). The CLI commands can vary based on the CLI in use (menu driven CLI or Cisco-like isCLI) and version of code. The following commands are used to determine the light level of an optical module that uses isCLI syntax:

- On older versions of code, the command **`show interface transceiver`** (and **`show transceiver`**) shows all SFP/SFP+/QSFP+ ports and any associated transceivers light levels that are rated in uW (micro watts). The output that is shown in Example 3-1 is an example of the use of the original format of the command to see the light levels for several SR SFP+ modules that are installed a Flex System SI4093 switch module.

Example 3-1 Retrieving light level by using original format isCLI syntax

```
SI4093#show interface transceiver

Name             TX  Link  TXFlt Volts DegsC TXuW  RXuW  Media   WavLen Approval
---------------- --- ----- ----- ----- ----- ----- ----- ------- ------ ----------
43 SFP+  1       Ena LINK  no     3.28 36.0  587.3 580.9 SR SFP+ 850nm  Approved
         IBM-Avago        Part:46C3448-L80181K  Date:140205 S/N:Y251UC41FDD1
44 SFP+  2       Ena LINK  no     3.28 36.0  580.9 665.0 SR SFP+ 850nm  Approved
         IBM-Avago        Part:46C3448-L80181K  Date:140111 S/N:Y251UC41BDKP
45 SFP+  3       Ena LINK  no     3.28 35.5  584.5 568.1 SR SFP+ 850nm  Approved
         IBM-Avago        Part:46C3448-L80181K  Date:140204 S/N:Y251UC41FDWO
46 SFP+  4       Ena LINK  no     3.28 35.5  588.2 560.7 SR SFP+ 850nm  Approved
         IBM-Avago        Part:46C3448-L80181K  Date:140111 S/N:Y251UC41BD93
47 SFP+  5       < NO Device Installed >
48 SFP+  6       < NO Device Installed >
49 SFP+  7       < NO Device Installed >
50 SFP+  8       < NO Device Installed >
51 SFP+  9       < NO Device Installed >
52 SFP+ 10       < NO Device Installed >
```

- If you do not see the light levels in the output of the **show interface transceiver** command (and assuming there are optical modules that are installed and not DAC), you are using newer code that changed how light levels are viewed. In later code, the light level outputs can be seen on a port-by-port basis only and now uses the syntax **show interface port X transceiver details** (where X equals the port number to be queried). The output that is shown in Example 3-2 is an example of the use of the new style command to determine the light level for an SR SFP+ module that is installed in port 17 in a G8264CS switch.

Example 3-2 Retrieving light level by using new style isCLI syntax

```
G8264CS#show int port 17 transceiver details

Port             TX  Link  TXFlt Volts DegsC TXuW  RXuW  Transceiver    Approve
---------------- --- ----- ----- ----- ----- ----- ----- -------------- --------
17 SFP+  1       Ena LINK  NoFlt  3.29 26.5  585.9 486.4 SR SFP+        Approved
         IBM-Avago        Part:46C3448-L80181K  Date:140110 S/N:Y251UC41AD20
```

3.2 Considerations for low-level interface configurations

This section describes the low-level commands that can be applied to interfaces that affect basic operation on the physical medium.

3.2.1 Speed, duplex, and auto negotiation settings

All built-in (data and management) RJ45 ports support different speeds and duplex. This support includes negotiating 10 Mb, 100 Mb or 1000 Mb (1 Gb), and full or half duplex, in addition to independently enabling or disabling auto negotiation of these features.

One way in which Lenovo switches vary from Cisco switches is what happens to auto negotiation when a port is hardcoded for speed and duplex. When you hardcode speed and duplex on a Cisco switch, it also automatically disables auto negotiation. With Lenovo switches, hardcoding speed and duplex does not disable auto negotiation (it means that it attempts to auto negotiate with the other side *only* for the speed/duplex that is hard set and still continue to negotiate other elements that are part of auto negotiation, such as auto-MDI/MDI-X and Flow Control). To fully disable auto negotiation on a Lenovo port, the command **no auto** must be run on that port. Before you set a port to **no auto**, the speed and duplex must first be set on that interface.

Ports that are running 1 Gb and faster should auto negotiate. The reason is that auto negotiation negotiates speed, duplex, and flow control support (pause frames) and auto MDI/MDI-X (auto straight-through or cross over cable detection). Therefore, if auto negotiation is disabled, the type of cable (straight-through or cross over) becomes a possible issue. With auto negotiation enabled, you can use either cable type and it automatically adapts to that cable. With auto negotiation disabled, if the switch port is connecting to another switch port, it *must* be a cross over. If the switch port is connecting to an end device (for example, a host NIC or router), it must be a straight through cable.

If 10/100 operation is wanted, you must decide whether to hardcode speed and duplex. This decision is mostly based on the customer standard is for these 10 Mb or 100 Mb connections.

For Lenovo 1 Gb ports that are part of an installable module (SFP), these ports are only 1 Gb and cannot be set to 10 Mb or 100 Mb.

For RJ45 10 Gb ports (on G8264T), these ports support 100 Mb, 1 Gb, or 10 Gb only (no 10 Mb), and default to auto negotiation. Only full duplex operation is supported at 1 Gb and 10 Gb speeds.

One common issue that is encountered for auto negotiation is if one side of a 1 Gb or lower connection is configured for auto negotiation and the other side has auto negotiation that is disabled and hardcoded for a specific speed and full duplex. In this case, the side set for auto negotiation often goes to half duplex, which leads to a duplex mismatch and subsequent performance issues and errors on the link. To prevent this issue, *always* use the same negotiation on both sides of a link (enabled on both sides, or disabled on both sides, and if hardcoded, set to the same speed and duplex).

3.2.2 Flow control

Flow control is also a feature that can be hardcoded or negotiated. By default, most 10 Gb and 40 Gb ports on Lenovo switches have flow control that is disabled (some Lenovo embedded switches default their 10 Gb server-facing ports to flow control that is enabled, but the uplink 10 Gb and 40 Gb ports for these switches have default settings to disable flow control).

Flow control can be enabled on these interfaces; however, it is not a good idea. In particular, avoid flow control on most links between networking devices that might carry many conversations from many end devices.

This practice (flow control that is disabled between network devices) is true for most vendors, not only Lenovo. The reason is that a pause frame (used in flow control) that is received on a port tells that port to stop allowing packets for some time (defined in the pause frame), and it affects *all* incoming packets (except other pause frames). Therefore, if an upstream switch receives a pause frame that is generated by a downstream device, it discards all traffic from *all* devices sending in to this link, not only a single distressed device that generated the pause frames downstream.

3.2.3 Jumbo Frame considerations

All Lenovo Networking switches support jumbo frames up to a minimum of 9 KB (9216 bytes) packet sizes, which is often referred to as maximum transmission unit (MTU), and nothing needs to be done to the switch to enable this jumbo frame support. This support is enabled and cannot be disabled. Some Lenovo switches also support frames that are larger than 9 KB.

Table 3-2 on page 43, Table 3-3 on page 43, and Table 3-4 on page 43 list which Lenovo switches support which size jumbo frame packets. Not all enclosure-based switches for the Flex System or BladeCenter chassis are Lenovo branded; therefore, those switches have their own vendor-specific limitations and are not included in these tables.

Important: Although jumbo frames are supported by default on all data interfaces on Lenovo switches, jumbo frames are not supported on any management interface on these switches. Also, jumbo frames are not supported on most Lenovo switches IP management interfaces. If you ping most Lenovo switches with a jumbo frame packet, it fails. The switch can pass these packets correctly, but the management of the switch does not support jumbo frames in a conversation that involves the switch processor.

Table 3-2 lists the maximum frame sizes for each RackSwitch top-of-rack switch model.

Table 3-2 MTU support for all Lenovo RackSwitch switches

Switch model	Max Jumbo/MTU frame size	Comments
RackSwitch G7028	12,288	
RackSwitch G7052	12,288	
RackSwitch G8000	9,216	
RackSwitch G8052	9,216	
RackSwitch G8124	9,216	
RackSwitch G8264	9,216	Same for all G8264 models
RackSwitch G8316	9,216	
RackSwitch G8332	9,216	

Table 3-3 lists the maximum frame sizes for each Flex System switch model.

Table 3-3 MTU support for all Lenovo Flex System switches

Switch Model	Max Jumbo/MTU frame size in Bytes	Comments
Flex System EN2092	9,216	
Flex System EN4091	N/A	Passthru; MTU controlled by connecting devices
Flex System SI4093	9,216	
Flex System EN4093R	9,216	Same for EN4093R and EN4093 model
Flex System CN4093	9,216	

Table 3-4 lists the maximum frame sizes for each BladeCenter switch model.

Table 3-4 MTU support for all BladeCenter switches

Switch Model	Max Jumbo/MTU frame size	Comments
Intelligent Copper Pass-Thru Module	9,216	
Server Connectivity Module	9,216	
Layer 2/3 Gigabit Ethernet Switch Module	9,216	
Layer 2-7 Gigabit Ethernet Switch Module	9,216	
1/10Gb Uplink Ethernet Switch Module	9,216	
10Gb Ethernet Pass-Thru Module	N/A	Passthru; MTU controlled by connecting devices
6-port 10 Gb Ethernet Switch Module	12,288	
Virtual Fabric 10Gb Switch Module	9,216	

To check whether a port is sending or receiving jumbo frames, the maintenance counters can be inspected on most switches; for example, by using the **show int port X maintenance-counter** command, as shown in Example 3-3.

Example 3-3 Edited output that shows frame size counters

```
G8264CS#show int port 17 maint
---------------------------------------------------------------------
Maintenance statistics for port 17:
<<<Snip>>>
Receive 64 Byte Frame Counter
HW: MAC_GRx64 : 481000
Receive 65 to 127 Byte Frame Counter
HW: MAC_GRx127 : 5761
Receive 128 to 255 Byte Frame Counter
HW: MAC_GRx255 : 17954
Receive 256 to 511 Byte Frame Counter
HW: MAC_GRx511 : 9224
Receive 512 to 1023 Byte Frame Counter
HW: MAC_GRx1023 : 0
Receive 1024 to 1518 Byte Frame Counter
HW: MAC_GRx1518 : 0
Receive 1519 to 1522 Byte Good VLAN Frame Counter
HW: MAC_GRx1522 : 0
Receive 1519 to 2047 Byte Frame Counter
HW: MAC_GRx2047 : 0
Receive 2048 to 4095 Byte Frame Counter
HW: MAC_GRx4095 : 0
Receive 4096 to 9216 Byte Frame Counter
HW: MAC_GRx9216 : 0
Receive 9217 to 16838 Byte Frame Counter
HW: MAC_GRx16383 : 0
<<<Snip>>>
Transmit 64 Byte Frame Counter
HW: MAC_GTx64 : 0
Transmit 65 to 127 Byte Frame Counter
HW: MAC_GTx127 : 0
Transmit 128 to 255 Byte Frame Counter
HW: MAC_GTx255 : 17605
Transmit 256 to 511 Byte Frame Counter
HW: MAC_GTx511 : 0
Transmit 512 to 1023 Byte Frame Counter
HW: MAC_GTx1023 : 0
Transmit 1024 to 1518 Byte Frame Counter
HW: MAC_GTx1518 : 0
Transmit 1519 to 1522 Byte Good VLAN Frame Counter
HW: MAC_GTx1522 : 0
Transmit 1519 to 2047 Byte Frame Counter
HW: MAC_GTx2047 : 0
Transmit 2048 to 4095 Byte Frame Counter
HW: MAC_GTx4095 : 0
Transmit 4096 to 9216 Byte Frame Counter
HW: MAC_GTx9216 : 0
Transmit 9217 to 16383 Byte Frame Counter
HW: MAC_GTx16383 : 0
<<<Snip>>>
```

As shown in Example 3-3 on page 44, `GR` represents receive counters and `GT` represents transmit counters for the port shown.

If there is ever any doubt if a switch is passing jumbo frames, these counters can be observed. If the receive counters are not seeing jumbo frames coming into a port, the device that is connecting to that port is not sending jumbo frames.

4

Layer 2 technologies

This chapter describes preferred practices for Layer 2 technologies and includes the following topics:

4.1 Virtual Link Aggregation Group considerations

This section introduces virtual Link Aggregation Groups (vLAGs) and describes considerations for operating switches in a vLAG environment.

4.1.1 Introduction to vLAG

vLAG is a feature on Lenovo Networking switches that allows a pair of switches to act as a single endpoint for an aggregation, and is similar in function to Cisco Virtual PortChannel (vPC). It provides improved high availability compared to a single switch that is acting as an endpoint and can enhance performance by splitting loads across the aggregated links and switches.

For more information about configuring the vLAG feature, see the Application Guide for your product. Although this chapter provides a simple explanation, for more information about is a vLAG is, see, *NIC Virtualization in Flex System Fabric Solutions*, SG24-8223, which is available at this website:

http://lenovopress.com/sg248223

4.1.2 Understanding packet flow in a vLAG environment

This section describes how traffic flows when vLAG is enabled; specifically, the potential for normal traffic to use the Inter Switch Links (ISLs) between a pair of vLAGed switches in certain designs. Although the explanation in this chapter is focused on embedded Flex System switches, it can also apply to connecting stand-alone servers to stand-alone switches that are running vLAG.

Related terms and concepts

The following related terms and concepts are important for understanding packet flow in this environment:

- *Link Aggregation (LAG)* is also referred to as PortChannel, Etherchannel, trunking, and other terms. This section uses the terms Aggregation or LAG to refer to the bundling of physical links to act as a single logical link. Figure 4-1 on page 49 shows an example of traditional LAG with a simple, four-port aggregation (the maximum number of ports that are supported in an aggregation is vendor and device specific). Aggregations with a Lenovo switch use standards body (IEEE 802.3AX LACP) or industry standards (for static aggregation) to ensure compatibility between vendors.

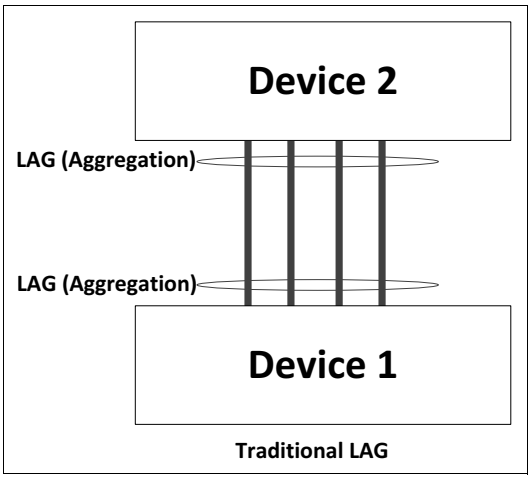


Figure 4-1 Example of a simple Link Aggregation

- *vLAG* is similar to Cisco vPC or Nortel Split Multi-Link Trunk (SMLT). Lenovo Networking vLAG is a form of multi-chassis (multi-switch) aggregation. vLAG is *not* a form of aggregation in its own right; instead, it is an enhancement to current aggregation standards. vLAG attempts to overcome a shortcoming of standards-based aggregations.

  By current standards definitions, an aggregation can connect two devices only (see "Traditional LAG" in Figure 4-1). In traditional LAG, if one device on either end of the aggregation fails, the entire path is gone. By using vLAG, you can take a pair of switches and make them logically act as a single switch for aggregation purposes (see the split LAG example in Figure 4-2). This example splits the aggregation on the end that is running vLAG (or both sides if the other side is also running some form of split LAG); therefore, a single switch failure in that pair does not take down the entire path.

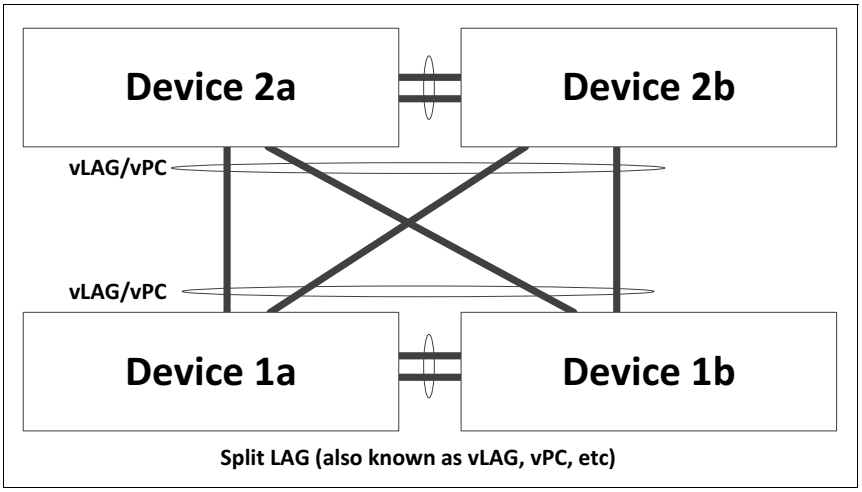


Figure 4-2 Example of cross switch aggregation with vLAG

- *Local preference* is used in multi-chassis aggregation environments (such as vLAG), where a single aggregation might have links to a single upstream path split over ports on two different switches. To understand the need for local preference in vLAG environments, it is helpful to first understand how packet flow works *without* local preference.

Without local preference, a packet on one switch in the vLAG pair that needed to get upstream uses normal aggregation hashing. It might use the *other* switch in the pair to get to the upstream network and add an unnecessary hop across the ISL before it heads upstream (the ISL connects the physical pair to help form the virtual pair). An example of this extra hop is shown in Figure 4-3.

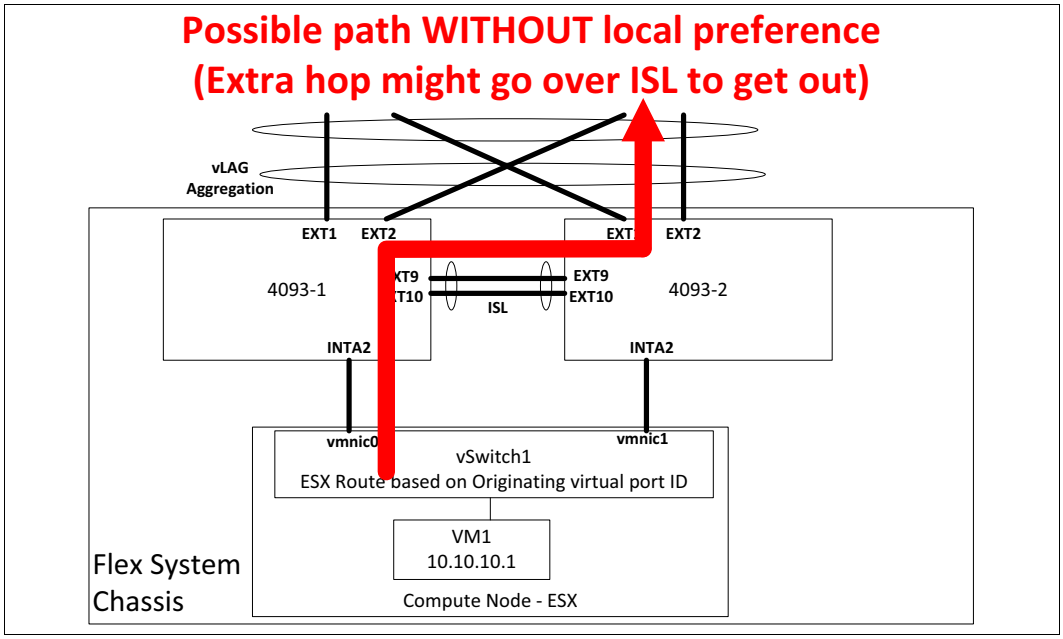


Figure 4-3 Without local preference, packets might take the ISL, even if local links are up

With local preference, if a packet on a switch in the pair needs to get to the upstream network and if that switch has *any* links in that common cross chassis aggregation toward the destination that is *up*, the switch always *prefers* the local links to send to the upstream, and does not add that extra hop. An example of this operation is shown in Figure 4-4 on page 51.

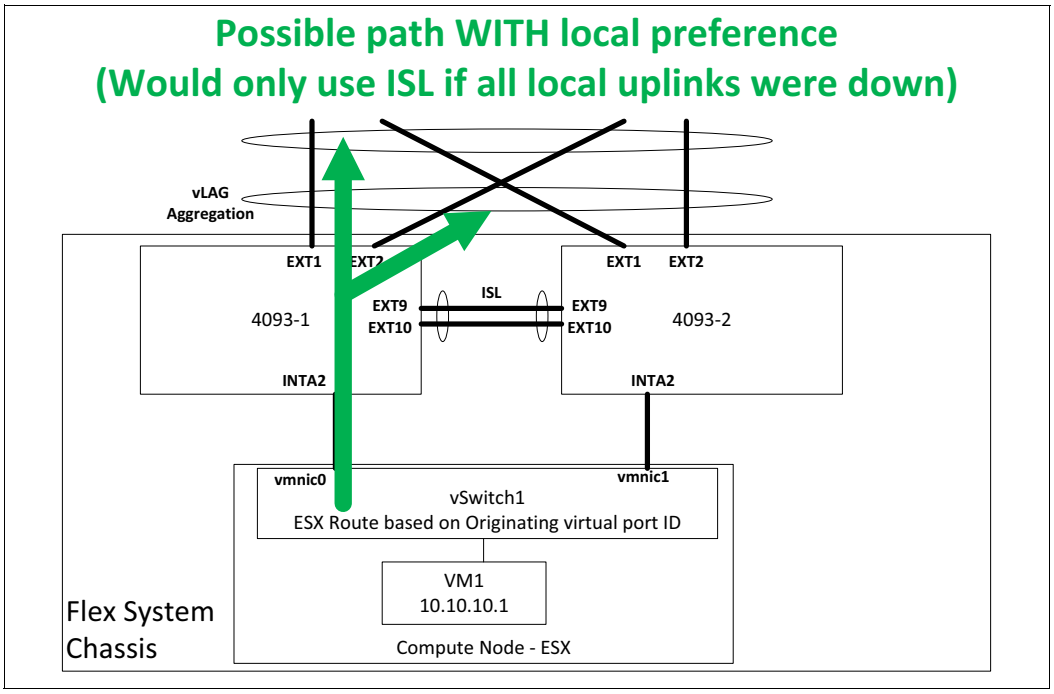


Figure 4-4 With local preference, local links take precedence over the hash

In addition to reducing hops, local preference also reduces the overall load on the ISL between the pair of vLAGed switches, which might permit having a smaller ISL than if local preference is not used. Local preference is built into vLAG and cannot be disabled. All versions of vLAG support local preference, but not all vendors support it in their implementations of cross-chassis aggregation technologies.

- *Switch independent mode teaming and switch dependent mode teaming* (which is also known as bonding in some operating systems) is a method of combining NICs in the server to increase performance and high availability. Some of these teaming modes can affect how the upstream switches to the server must be configured and how traffic flows through a vLAGed environment. Switch independent mode teaming does not require configuration on the switch (except for allowing the specified VLANs). Switch-dependent mode teaming requires a form of special configuration on the switch (in the form of aggregation) to interoperate with the teaming mode on the server.

For more information about teaming modes and their interaction and operation, see *NIC Virtualization in Flex System Fabric Solutions*, SG24-8223, which is available at this website:

`http://lenovopress.com/sg248223`

The remaining parts of this section describe the examples that are shown in Figure 4-5 on page 52 with switch-independent mode teaming, and Figure 4-6 on page 54 with switch-dependent mode teaming to help show how packets can flow in an environment that features vLAG.

Packet flow with switch-independent mode teaming

The example that is shown in Figure 4-5 is described in this section, via step-by-step numbered packet flows.

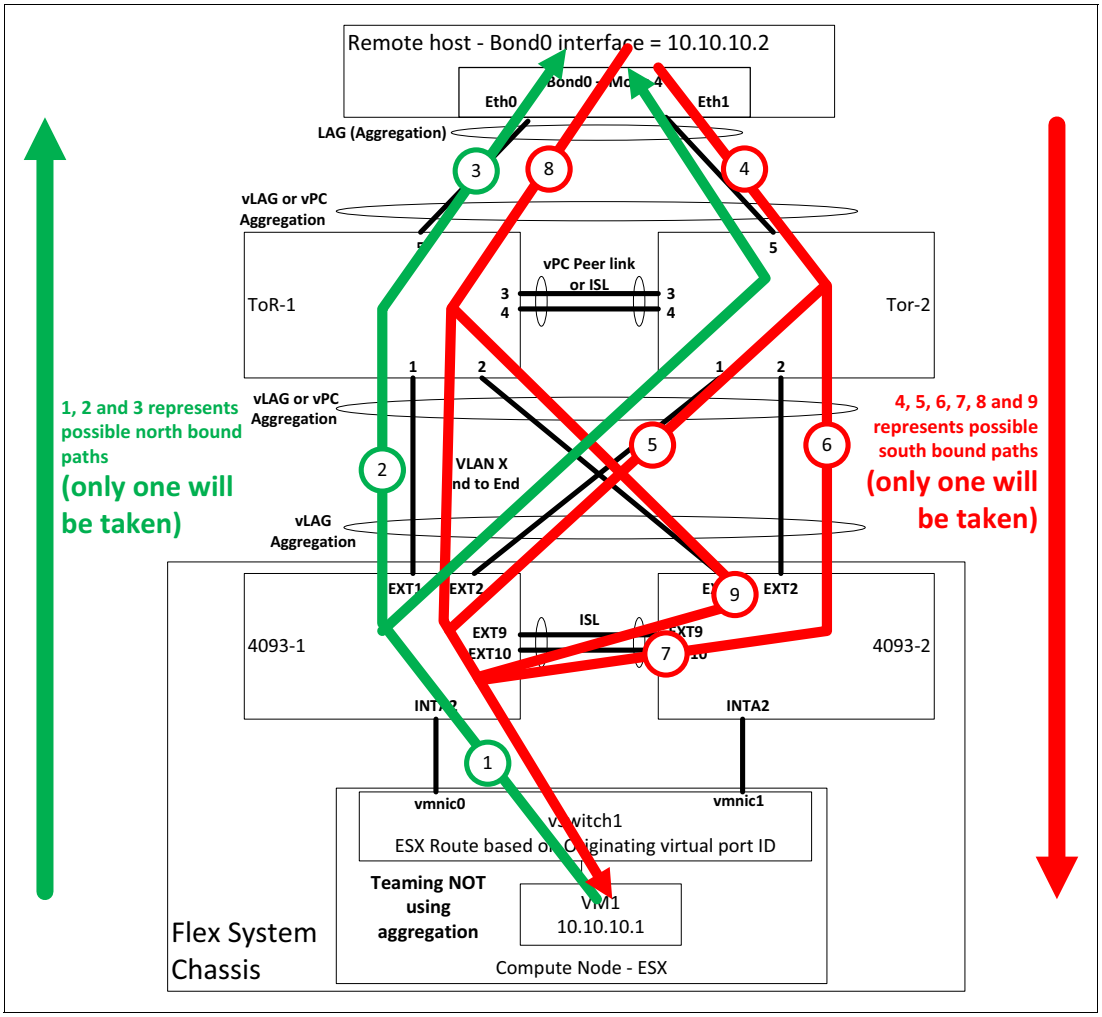


Figure 4-5 Possible packet flows with a host that uses a non-aggregation form of teaming

This packet flow includes the following steps, as shown in Figure 4-5:

1. In this example, the vSwitch in the local host is not running an aggregation-based teaming; instead, it uses load balance per virtual machine (VM) (this setting is the default setting in an ESX vSwitch). The vSwitch for VM1 uses only one of the vmnics (that is, the network interface cards) for all conversations for VM1 into and out of the ESX host. For this example, assume that it hashed to using `vmnic0` to the left side for VM1 (it might as well decide to use `vmnic1`, but whatever vmnic the vSwitch does hash VM1 to, it sends packets from VM1 only out that direction, and accepts packets that are destined for VM1 only back on that same vmnic if vmnic stays up).
2. VM1 pings the remote host for the first time. Therefore, an Address Resolution Protocol (ARP) broadcast is sent to 4093-1. When you are using local preference, 4093-1 always sends that packet over one of its own uplinks (if a local uplink is available), *not* the ISL. For this example, assume that it chose EXT1 toward ToR-1 (although it might also use EXT2).
3. The packet gets to ToR-1 and ToR-1 uses its own hash to send it to the remote host. If ToR-1 has local preference, it uses port 5 on ToR-1 to send the packet directly to the remote host that is using aggregation and it comes in on eth0 on the remote host.

4. Remote host responds. As far as the remote host is concerned, either interface (`eth0` and `eth1`, part of a bond mode 4 [LACP aggregation] interface named bond0) is as usable as the other, and it uses its own hash to decide the return path. Assume that it picked `eth1` for the response, so the return packet is sent to ToR-2. If ToR-2 also has local preference, it can choose to send that response out port 1 or 2, and it uses its own hash to decide.
5. If ToR-2 decides port 1, that packet goes to 4093-1 and takes the same interface (`vmnic0`) back into the host.
6. If ToR-2 decides to use port 2 to send that response, the packet comes in on 4093-2.
7. VM1 has no active interface on 4093-2, and VM1's MAC is known *only* on port INTA1 on 4093-1, *not* on INTA1 on 4093-2. If the VLAN that is used is carried across the ISL (which it should be), the MAC table for 4093-2 also has the MAC for VM1 point to the ISL, and in this case, the return packet *must* cross the ISL to get back to VM1.
8. The same situation might occur if the remote host decided to respond on `eth0` instead and sent it toward ToR-1.
9. ToR-1 also uses local preference and does *not* use the ISL; however, it *can* select port 1 or port 2. If it selects port 2, the packet goes to 4093-2 and then *must* cross the ISL 4093-2 - 4093-1 to complete the path back to VM1.

Figure 4-5 on page 52 shows how the possible outbound packets might flow north (items 1 - 3), with this switch-independent mode teaming design, and the possible return paths the packets might take are shown in items 4 - 9. (It is assumed that all links are up.) All possible paths also are shown, but it is assumed that only one of these paths is taken.

Consider the following points when switch independent mode teaming with vLAG is used:

- There are designs within which normal traffic can use the ISL (not only low-level vLAG traffic and failover traffic). Figure 4-5 on page 52 shows this ISL path usage on the return traffic.
- You *must* carry all VLANS on the ISL that goes down to the hosts and uplinks in a common aggregation. If you do not carry all VLANs on the ISL, a packet is discarded at best when it gets back to 4093-2 or it is flooded to all ports (and even if it goes to INTA1 on 4093-2, the vSwitch on the host discards the packet because it is using *only* the `vmnic0` toward 4093-1 for that VM) and you create a black hole for traffic.

  The ToR switches that are shown in Figure 4-5 on page 52 do not know that there are potentially two separate switches below them. Instead, 4093-1 and 4093-2 appear to be a single switch and *any* port is as good as another to get traffic from the ToR switches to the 4093 switches, so the network administrators *must* ensure that the proper VLAN paths exist when vLAG and vPC are used.
- During normal operation (all links are in an up state), only *returning* inbound traffic to the local host might need to use the ISL in this design (depending on what path the ToR used). Outbound traffic from the local host in this design *always* uses the local uplinks to get out (based on local preference) and do *not* cross the ISL.
- Although this path is not shown in this example, the ISL might be used in other normal circumstances. For example, if two VMs on different ESX hosts in this same Flex System chassis hash to different I/O modules for outbound traffic (specific to the use of a switch-independent mode of teaming, such as the VMware route that is based on the originating virtual port ID). In that case, any communications between those two VMs goes over the ISL links.

Packet flow with switch-dependent mode teaming

The next example shows the hop-by-hop packet flow between a host that is running switch-dependent mode teaming in the Flex System chassis and the aggregated host at the top of the design. The numbered steps correspond to the numbers that are shown in Figure 4-6.

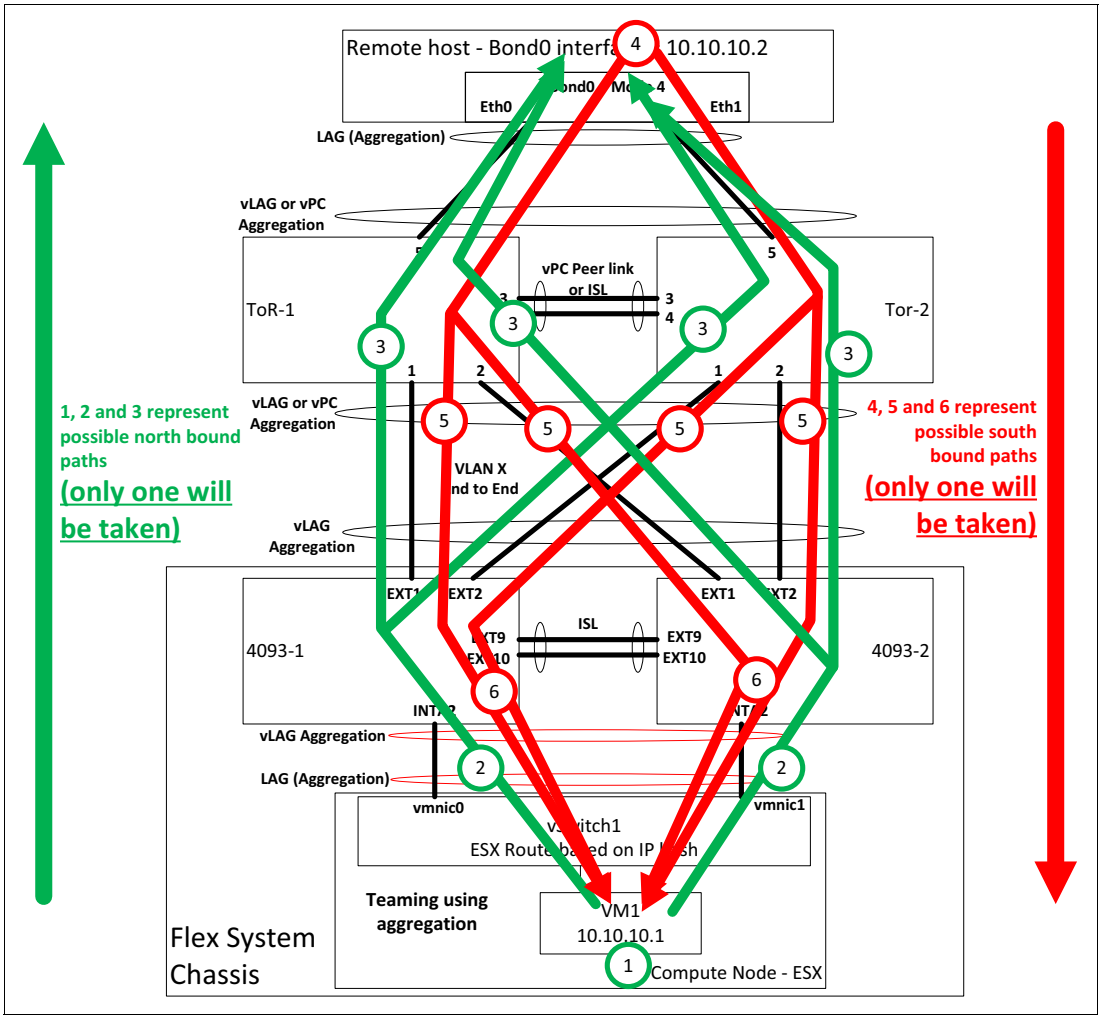


Figure 4-6 Possible packet flows with host that uses an aggregation form of teaming

This packet flow includes the following steps, as shown in Figure 4-6:

1. The vSwitch in the local host is running a form of aggregation (switch-dependent mode of teaming) and because of this configuration, the ports on the 4093 switches that are facing the internal host also are running a form of aggregation. The vSwitch for VM1 picks one link to send packets for a session out of that VM by using the vSwitch's own hashing strategy. That vmnic varies, depending on the conversation (it can use vmnic outbound for a specific conversation, but uses only that one outbound path for that conversation).
2. VM1 pings the remote host. An ARP broadcast is sent, which goes up to 4093-1 *or* 4093-2 (not both), based on the vSwitch aggregation hash decision. Whichever switch it is sent to, that 4093 uses *only* its EXT1 or EXT2 ports to forward the packet, *not* the ISL (unless all EXT ports are down on that 4093 switch) to get to the next hop (ToR switches).
3. The packet is sent to one of the ToR switches and the ToR (assuming local preference on the ToR switches) uses its local port to forward it toward the remote host.

4. The remote host responds. As in the previous example, for the remote host, either interface is usable and it uses its own hash to decide return NIC (as with all aggregation paths, it picks *one* path for a specific conversation).
5. The return packet can be to ToR-1 or ToR-2. Assuming that they both have local preference, ToR-1 or ToR-2 can choose to send that response out port 1 or 2, and uses its own hash to decide.
6. If ToR-1 or ToR-2 chooses port 1, that packet goes to 4093-1 and takes the same interface (vmnic0) back into the host. However, if ToR-1 or ToR-2 decides to use port 2 to send that response, that packet comes in on 4093-2. Unlike the previous example in which the return packet crossed the ISL to get back to VM1, it can now use the local INTA2 interface on 4093-2 to send the packet directly to VM1. This local selection occurs because the vSwitch on the local host is using aggregation and 4093-2 uses local preference to send that packet directly to the host via vmnic1, *not* the ISL as it did in the previous example. In this case, the MAC table for 4093-1 and 4093-2 have an entry for the PortChannel interface that is facing the local host (and not the physical interface on 4093-1 as it did in the previous example) and thus both 4093 switches know that to get to the local host VM1, they can use their local path that is part of that split aggregation to that host.
7. Figure 4-6 on page 54 shows how the possible outbound packets might flow north (items 1 - 3) with this switch-dependent and aggregation mode of host teaming. The possible return paths the packets might take shown in items 4 - 6. This scenario assumes that all links are up. It also shows all possible paths, but assumes that only one of these paths is taken.

The example as shown in Figure 4-6 on page 54 also shows the following important items when switch dependent modes of teaming are used:

- Assuming that all local hosts are in an aggregation, the ISLs in this design are used for low-level vLAG traffic and fault recovery traffic only (that is, one of the 4093 switches loses all its uplinks); however, normal traffic does not need to use the ISL.
- You still *must* carry all VLANS on the ISL that also go down to the hosts to account for failover scenarios.
- This design is potentially more efficient for return traffic than the design in Figure 4-5 on page 52 because it can reduce hops for returning packets and reduce the potential load on the ISL.
- Although not shown in this example, any time different VMs on different ESX hosts inside this same Flex System chassis communicate with each other, that traffic never needs to cross the ISL (because local preference keeps traffic that is local to a switch in the pair) in normal conditions. This configuration is another improvement of network utilization over the switch-independent mode of teaming.
- Although this example shows that switch-dependent teaming operation is more efficient than switch independent teaming operation, this more efficient design is not always the best choice. Some examples where it might not be a better choice include the following circumstances:
  - For local hosts that are not using any form of teaming, using a switch-independent mode of teaming, or are single-homed into one of the pairs of switches only, you cannot use this type of aggregated connection option to the server.
  - For environments that do not support vLAG, stacking, or some other form of multi-chassis aggregation to the embedded hosts in the Flex System chassis, you cannot use this design (not all forms of stacking support local preference, but stacking provides a multi-chassis aggregation, which is beyond the scope of this section).

– For environments that require LACP to the host and where some form of communications must take place to the local host *before* the operating system fully starts so LACP can start (for example, a fresh installation or bare-metal Preboot Execution Environment [PXE] boot), you cannot use this design (the 4093 switches do not pass traffic on a vLAGed LACP port if LACP is not formed). This issue also is a potential issue with the use of static aggregations to hosts that need PXE boot because before the operating system is loaded, the switch thinks it is one common static aggregation to the host. However, the host (before the operating system load) does not know the switch is in a static aggregation and is expecting any response to come back on the single interface on which it is sending out.

Note: These principles also apply to non-Flex System environments. The principles also apply to non-Flex System environments, such as stand-alone servers that connect to top-of-rack switches and provide the same functions for the server connectivity.

4.1.3 Understanding vLAG Tier IDs

The Tier ID is a mandatory configuration for vLAG. This Tier ID must be the same on a pair of vLAGed switches, but unique between pairs of connecting vLAGed switches. As shown in Figure 4-2 on page 49, switches 1A and 1B use a common Tier ID. Switches 2A and 2B also use their own common Tier ID, but it is different from the Tier ID that is used by switches 1A and 1B.

The reason connecting vLAGed pairs must use a unique Tier ID is that this Tier ID is used to generate a common shared MAC, so that the other side of the vLAG pair thinks it is aggregating with a single device. If two pairs of vLAGed switches (four switches total) are connected with the same Tier ID, they attempt to use the same MAC, which does not work.

This Tier ID generated MAC is derived from a base from a Lenovo reserved range of 08:17:f4:c3:dd:00 - 08:17:f4:c3:de:ff.

The last two bytes are determined by the vLAG Tier ID setting, as shown in Example 4-1.

Example 4-1 Example vLAG Tier ID settings

```
vLAG tier ID 1 MAC = 08:17:f4:c3:dd:00
vLAG tier ID 2 MAC = 08:17:f4:c3:dd:01
vLAG tier ID 512 MAC = 08:17:f4:c3:de:ff
```

To show the MAC address that is generated by the Tier ID, use the **show vlag info** command.

This MAC address is not used by Virtual Router Redundancy Protocol (VRRP) or for any other functions or communications to end hosts (it is used only to form cross-switch aggregations and to present a common MAC between the vLAGed pair to the other side of the cross-switch aggregation).

If it is using non-vLAG LACP aggregation, the local aggregation uses a MAC from the base system MACs that are available on a switch (as displayed by the **show system** command) to form the aggregation.

4.1.4 Importance of a proper health check network with vLAG

An optional configuration for vLAG is a health check network. Without a health check network configured, the ISL is the only way a vLAGed pair knows that the other switch in the pair is up and in what state. If the ISL goes down without a separate health check network configured, the switches do not know whether the other switch is down or only the ISL path is down. In this state, both switches go active forwarding without knowing what the other switch is doing, and this situation can cause issues with connectivity for attaching devices.

To prevent this situation, an optional health check network can be configured. Owing to the critical function the health check configuration provides, it should be considered mandatory for any production environments.

Consider the following rules for configuring a health check network:

- An IP interface must be configured on each switch in the pair.

  This IP interface can be shared for health check and switch management, but a dedicated health check IP interface is preferred.

  For embedded switches that support vLAG, this IP interface *cannot* be the IP interface that is provided by the embedded environment's management module; for example, the IP address and interface that is assigned by a Chassis Management Module (CMM) in a Flex System environment cannot be used for the vLAG health check network.
- Do not configure the IP interfaces for health check to use the ISL links to communicate between each other. In this case, if the ISL goes down, so does the health check network and the split brain vLAGed pair can still occur. A separate health check network is designed to prevent this situation.
- The IP interfaces that are used for this purpose cannot connect to each other over an aggregation *with* vLAG. Because the split aggregation of a vLAG pair is considered a single logical interface, any health check packet that goes out the vLAGed aggregation cannot come back to the other switch with the same vLAG aggregation to complete the health check path.

  One exception to this rule is if the health check IP interfaces are on different IP subnets. In this situation, they can be routed, but it is best to configure the health check on a common IP subnet to minimize unexpected disruptions to the health check path.
- If the ISL is up, the health check network status has no affect on operation. Only if the ISL goes down does the health check network play a part. The following states of the combination of the ISL and health check network status are possible:
  - Health check is up and the ISL is up. The vLAG packet passing is 100% operational.
  - Health check is down and the ISL is up. The vLAG packet passing is 100% operational, but there is no protection from an ISL failure if the ISL goes down.
  - Health check is down and the ISL is down. Split brain operation occurs and both switches are forwarding. This failure of two components is considered a double fault, which often is not considered when redundant designs are built.
  - Health check is up and the ISL is down. This situation is why the health check network is important. Consider the following points:
    - In this case, the primary VLAG member brings all links forwarding and the secondary vLAG member error-disables all aggregations that are configured to use vLAG.
    - This configuration is critical to ensure a stable environment when an ISL failure occurs.

- When the ISL goes back up, the switches automatically return to normal vLAG operation after a brief time to ensure stability.

4.1.5 ISL considerations

The ISL is a special and important component for a vLAG environment and has the following unique requirements:

- The ISL must be some form of aggregation (LACP or static). In general, LACP is preferred, owing to the nature of LACP to protect from misconfiguration or miscabling. However, static aggregations work for this purpose.
- The ISL should always be a minimum of two physical links.

  A single link aggregation can be configured and used, but it is not advised because it becomes a single point of failure for the vLAG.

  It is also possible to take a 40 Gb port, set it to 4 x 10 Gb mode, set an aggregation across that, and use a single QSFP+-to-QSFP+ cable to carry that aggregation. This configuration is also not advised because, although logically it is four different 10 Gb ports, it is using a single physical cable and that single cable becomes a single point of failure.
- Although it is possible to use 1 Gb ports for the ISL and that setting is okay for switches that are primarily 1 Gb (for example, a G8052), use 10 Gb or 40 Gb links for this ISL aggregation to ensure proper performance if the switch is primarily a 10 Gb/40 Gb switch (for example, a G8264 or EN4093).
- Set an unused VLAN (not carried to any other ports) as the native or PVID VLAN on the ISL links and disable Spanning Tree Protocol (STP) for this VLAN (assuming that STP is not globally disabled).

  The use of this configuration helps to ensure that if one switch or the other is returned to factory default settings, it cannot create a loop between switches. It helps to prevent this issue because the defaulted switch is sending untagged packets *only*, and those untagged packets are sent into an unused VLAN on the side that was not returned to factory default settings. Also, no loop is created.
- Sizing of the ISL path is not a simple topic because it depends on the number and speed of uplinks and downlinks, the host bandwidth requirements, and how the hosts are configured for teaming (as described in 4.1.2, “Understanding packet flow in a vLAG environment” on page 48).

  One method for sizing the ISL is create the ISL aggregation to be equal to 50% of the primary vLAG aggregation uplinks out of a specific vLAGed pair. For example, if a pair of EN4093s each has 4 x 10 Gb uplinks that create an 8 x 10 Gb vLAG aggregation that is headed upstream, create a 4 x 10 Gb ISL (50% of the total uplink capacity). The logic is that if one side loses all uplinks, it has an equal size path over to the partner switch of the vLAGed pair through the ISL. Consider the following points:

  – This 50% might be considered excessive because you can saturate the uplinks of the other switch, but it depends on how much bandwidth is normally in use.

  – The numbers that are provided in this section are limited by the total number of available uplinks for use for vLAGed aggregation uplinks and the ISL aggregation; therefore, it might not be practical for every environment.

4.1.6 Other considerations for vLAG

Consider the following points regarding vLAG:

- When you are using vLAG, mono-instance RSTP is not supported.
- Owing to the fact that vLAG often is used to create non-looped designs; it is not uncommon to disable spanning-tree globally on a vLAGed pair of switches.
- If spanning tree is required, PVRST (default) and MSTP can be used on the vLAG pairs. A switch with Spanning Tree disabled supports more vLAGed aggregations than a switch that is running Spanning Tree enabled. For more information about the specific product for vLAG limitations that are related to Spanning Tree, see the Application Guide.
- When you are using PVRST, it is important to keep the instances of STP that are assigned to VLAN in sync between switches in a vLAG pair. (Matching instances is for proper operation for MSTP in any environment. However, for PVRST, it does not matter with non-vLAGed switches, whereas it does matter between a pair of vLAGed switches).

  If the STP-instance-number-to-VLAN is out of sync between a pair of vLAGed switches, odd spanning-tree blocking might occur. For example, a **`show span block`** command might show that the ports of a vLAGed aggregation are blocked on one switch, but not on the partner switch. This situation should never occur because it is a single aggregation (split across only two switches).

  If the STP-instance-number-to-VLAN assignment is out of sync between a pair of vLAGed switches, manually correct this configuration by manually getting them into sync to prevent unexpected communications issues.
- When you are looking at spanning tree (for example, by using the **`show span`** command) on a vLAGed pair of switches and the other side of the vLAG aggregation is towards the root, the root bridge appears as being on the uplinks of the vLAG primary switch of the pair; however, the secondary switch appears as both on the uplinks and toward the ISL of the other vLAGed switch. This display is not an indication of an issue. Instead, it is an artifact of how vLAG works and it does not affect operation or forwarding and blocking.
- Check the Application Guide for the model of switch and version of code in use to see whether there are any specific limitations with vLAG for that release and model.
- When you are using vLAG in a tiered design and the other side is another pair of vLAG switches, vPC, or some form of cross chassis aggregation, an optimal design is to connect at least one link between all four switches in the pair. This configuration is shown in Figure 4-2 on page 49, where each of the lower switches has a connection to each of both of the upper switches. The alternative is a design where the four links do not cross connect as shown. Both designs work, but the crossed design is more robust during switch failure events,
- Older code required configuring a special ISL VLAN. This configuration is not necessary on newer code. If a switch that is running older code is upgraded, the ISL VLAN setting command is removed automatically during the upgrade.
- When you are upgrading a vLAGed pair of switches, consider the following guidelines:
  - Upgrade and reload the primary switch of the vLAG pair first. During that reload, the secondary switch that is running the older code becomes the primary. After the primary is fully operational and forwarding, upgrade and reload the remaining switch in the vLAGed pair. Use the **`show vlag info`** command to determine which switch is acting as *primary*.
  - When you are upgrading vLAGed pairs, it is important to have both switches running the same version of code. Do not upgrade one switch in the pair and leave the switches running for an extended time on different versions of code.

4.2 Stacking

This section describes the stacking feature that is available on the following switch models:

- BladeCenter Virtual Fabric Switch
- G8000
- G8052
- G8264
- EN4093
- CN4093
- EN/CN4093 hybrid stack

Stacking is not currently available on the following switches:

- G8124
- G8264T
- G8264CS
- G8316
- G8332

The stacking feature enables multiple switches or switch modules to operate as a single switch in the network. The stack has a single configuration, management address, forwarding database (FDB), and ARP table.

Each product has a list of features that are not supported when the product is used in stacking mode, which is published in the Application Guide for the product. Key features that are not supported include Layer 3 routing and sFlow monitoring.

Benefits and limitations of stacking

Stacking has the following benefits:

- The ability to operate multiple switches as a unit from a management perspective, with a single configuration file and a single management address for access (including SNMP).
- The ability to use ports from different physical switching elements as part of a single aggregation group. In chassis (BladeCenter, Pure System), this ability allows the pairs of switch ports that support a single server to be aggregated. The server can then use active/active aggregation that is based NIC teaming modes.

Stacking has the following limitations:

- When a stack is rebooted to activate new firmware or for any other reason, the entire stack (all of the switching elements) is rebooted at the same time. In a chassis-based design, this process typically disables all network access to the servers in the chassis for a time.
- Stacking does not support local preference. This lack of support causes some traffic to be needlessly forwarded around the stack links to get to the port that is chosen by the hashing metric when links are aggregated. This process is in contrast to vLAG, which does support local preference.

Considerations for stacking in a chassis

In an environment that consists of more than one chassis (BladeCenter or Flex System), there are multiple options to consider for deployment of stacking. Each option has its own benefits and limitations. The following options are available (assume that there are two switches in each chassis):

- The entire environment can be included in one stack. This configuration allows active/active switch-dependent mode NIC teaming (aggregation) on the servers but disables all network access to the servers when the stack is rebooted.
- Two stacks can be used. One stack consists of all the switches in I/O bay 1 of the chassis, and the other stack consists of all the switches in I/O bay 2. This configuration reduces the number of manageable network elements compared to unstacked environments and allows sharing and aggregation of uplink ports within a stack. However, it does not enable active/active switch-dependent mode (aggregation) NIC teaming. The two stacks can be rebooted one at a time so that the servers are not isolated from the network at any point.
- vLAG is not supported in stacked mode, which precludes the possibility of using the vLAG and stacking together to enable active/active switch-dependent mode teaming.
- Each chassis can have its own stack. This approach might be the best stacking option. It allows active/active teaming. It also allows uplinks to be aggregated and shared across two (or more) switches in the chassis and the environment can be upgraded and rebooted one chassis at a time. The only detriment to this option is that each chassis has a distinct switch configuration and is managed separately.

Figure 4-7 and Figure 4-8 show the stacking options. Figure 4-7 shows three chassis with two independent stacks.

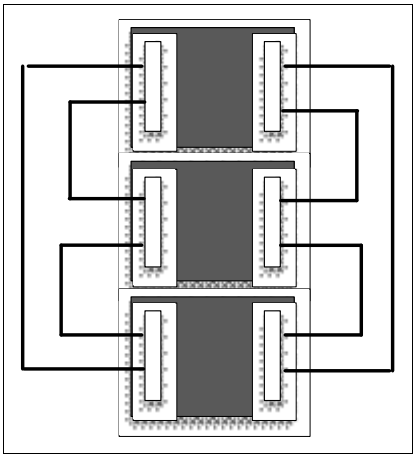


Figure 4-7 Three chassis with two independent stacks

Figure 4-8 shows three chassis with six embedded switches in a single stack.

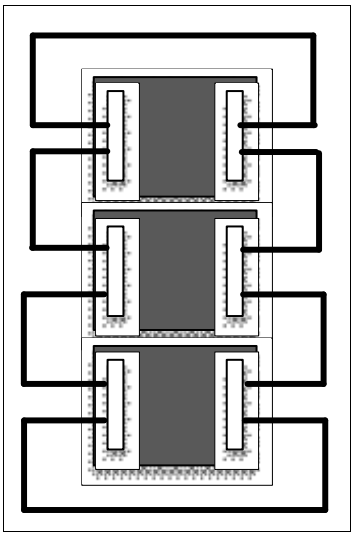


Figure 4-8 Three chassis with six embedded switches in a single stack

More stacking considerations

Consider the following guidelines for stacking:

- In stack mode on embedded switches, the port aliases (INTxx and EXTx) are not available. Only port numbers are shown. The following ranges for the port numbers are different in BladeCenter and Flex System chassis:
  - BladeCenter: INT1 - 14 is 1 - 14; Management is 15 - 16; EXT1 - 10 is 17 - 26
  - Flex System: INTA1 - 14 is 1 - 14; INTB1 - 14 is 15 - 28; INTC1 - 14 is 29 - 42; EXT ports are 43 - 64; Management ports are 65 - 66
- Several **`show`** commands in stack mode show all possible ports up to the maximum number of switches that can be attached to the stack, including switches that are not configured. Various command options can be used to limit the volume of output, including **`show int ... switch <n>`**.
- The stack links are assigned by default to VLAN 4090. This setting can be changed to a different VLAN, but this change must be done on each switch before the stack is formed. If the different switches are pre-configured with different values for the stack VLAN, the stack does not form correctly. The **`boot stack vlan <x>`** command is used to configure this value.
- Stacking ports must be manually configured on each switch or switch module before the stack can form. The **`boot stack higig-trunk <ports>`** command is used to configure this setting.

4.3 VLAN considerations

Most Lenovo Networking switches support configuring the entire range (1 - 4094) of active VLANs at one time, but certain features might restrict the total number of active VLANs. The relationship to STP instances also must be considered. Consider the following restrictions and some other attributes for VLANs that are specific to Lenovo switches:

- Some older switches, primarily BladeCenter switches, do not support all 4095 VLANs configured at one time. For example, the BladeCenter L2/3 Ethernet Switch Module (ESM), the 1/10 Uplink ESM, the L2-7 ESM, and the 10G Virtual Fabric Switch Module support a maximum of 1024 active VLANs at one time only, even in the latest code releases.
- When you are using stacking (supported on some Lenovo switches) and some other features, there is often a reduced number of total VLANs supported. For more information about for that product to understand any VLAN limitations that are imposed, see the limitations for each feature in the Application Guide.
- Some Lenovo switches and features reserve VLANs and these VLANs cannot be used for user VLANs. In these cases, some of these VLANs are changeable by the user, and some cannot be changed from the default. Reserved VLANs are listed in the limitations section of the features in the Application Guide of the product. Table 4-1 lists some reserved VLANs based on features.

Table 4-1 Some common reserved VLANs

Feature	Reserved VLANs	Configurable?	Comments
Flex System Interconnect Fabric (SIF)	4090 4091	Yes Yes	Fabric VLAN Blackhole VLAN
SPAR	4081-4088	Yes	VLAN 4081 = SPAR group1, 4082 = SPAR group2, and so on to 4088
FCoE	1002	Yes	Industry standard
UFP	4002-4005	No	N/A
Stacking	4090	Yes	N/A
No switchport	Highest free	Yes and no	See the list of items after this table
Management	4095	No	See the list of items after this table

Consider the following points about the VLANs that are listed in Table 4-1:

- For the preceding restrictions (except for Management VLAN 4095), the VLANs can be used if that specific feature is not in use or is reconfigured to use a different VLAN.
- When you are using the **`no switchport`** command to convert an L2 port to a routed L3 port, the highest available free VLAN is dedicated to that port and unavailable for other use. Each port that has the **`no switchport`** command reserves one VLAN for this purpose.
- Most Lenovo switches, indicate that VLAN 4095 exists and is used on one or more of the out of band management ports. This VLAN is often shown when you run the **`show int trunk`** or **`show int info`** commands, and might resemble the output that is shown in Table 4-2 on page 64.

Example 4-2 VLAN 4095 as seen on an EN4093 switch

```
bay-1#show int trunk
Alias   Port Tag    Type    RMON Lrn Fld Openflow PVID     DESCRIPTION  VLAN(s)
             Trk                                 NVLAN
------- ---- --- ---------- ---- --- --- -------- ------ -------------
INTA1    1     n  Internal    d    e   e       d       1   INTA1          1
INTA2    2     n  Internal    d    e   e       d       1   INTA2          1
...
INTA14   14    n  Internal    d    e   e       d       1   INTA14         1
EXTM     65    n  Mgmt        d    e   e       d    4095   EXTM           4095
MGT1     66    y  Mgmt        d    e   e       d    4095   MGT1           4095
```

- Consider the following points regarding how VLAN 4095 is used in different environments with Lenovo switches:
  - For embedded BladeCenter switches, packets that are traveling internally to the MM and AMM can be tagged with VLAN 4095. This tagging is used internally only and no packets that are tagged for VLAN4095 leave the BladeCenter (via the MM and AMM uplinks or by any of the switch uplinks). VLAN 4095 tagged packets are also used on internal server-facing ports in the BladeCenter, for such things as Serial over LAN (SoL) communications between the MM and AMM and the internal servers.
  - For Flex System embedded switches, two interfaces (MGT1 and EXTM) appear as being set to VLAN 4095. In this case, the 4095 is a place holder to indicate isolation from the switch fabric (the data ports). These ports send out only untagged packets and expect to receive only untagged packets.
  - Stand-alone Lenovo top-of-rack switches have none, one, or two out-of-band management ports (depending on the model). Lenovo switches that have an out-of-band management port show that port as belonging to VLAN 4095. As with the embedded Flex System environment, this display is being used only as a place holder to show it that it has no fabric connection. It sends, and expects to receive, untagged packets only.
  - For any device that has more than one out-of-band management port that uses VLAN 4095, there is no connection between these ports (although the VLAN 4095 might make it appear there is). This method indicates isolation from the fabric and any other out-of-band management port.
- Although many VLANs can be configured, only 127 instances of customer usable STP are available when the switch is running in the PVRST mode of Spanning Tree (the default mode of STP on most Lenovo switches). Assuming that a switch is in PVRST mode, these instances are automatically allocated and deallocated as VLANs are created and deleted until the number of VLANs exceeds 127 total VLANs (from the entire VLAN range). When all 127 of the user-available instances are in use, and more VLANs are added, they are assigned to STP instance 1 (instance 1 is the only instance that allows more than 1 VLAN when in PVRST mode and acts as a common instance of STP for VLANS with instance 1). If a VLAN that was allocated an instance other than 1 is deleted, the associated instance of STP becomes available and is used when the next VLAN is created.

  Having multiple VLANs in instance 1 is not a compatibility issue with Cisco or other devices that have support for larger instances of STP. It affects only load balancing redundant paths for VLANs in the common instance of STP. All VLANs in instance 1 on a Lenovo switch have the same forwarding and blocking characteristics.
- Do not add over 500 VLANs to a port with a single command. Adding too many VLANs at one time can lead to temporary high processor utilization and possible unstable operation until the process completes. If it is necessary to add more than 500 VLANs, shut down the port and add them or add them in amounts that are smaller than 500 VLANs at a time.

4.4 Private VLANs

Private VLANs (PVLANs) are used to improve VLAN security by restricting which devices can communicate with each other within a VLAN. This restriction is made with a Layer 2 configuration where a primary VLAN is segmented into one or more secondary VLANs, as shown in Figure 4-9.

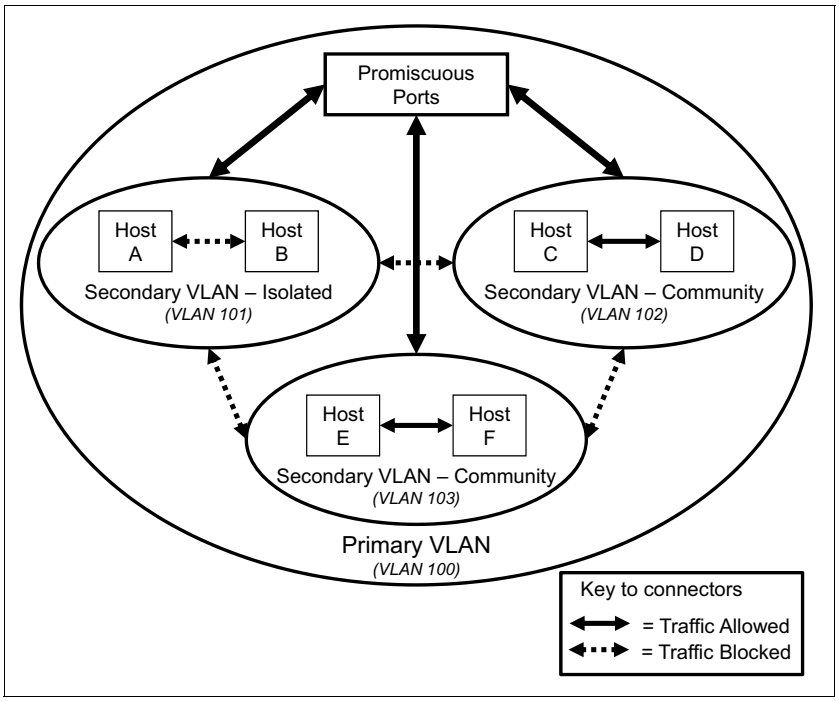


Figure 4-9 Layer 2 configuration

The primary VLAN is VLAN 100, which contains the promiscuous ports (P-Ports) and all of the secondary VLANs. Any device (or port) that is contained within the primary VLAN can communicate with any other device that is contained within the primary VLAN. Each promiscuous port can carry multiple VLANs when the port's mode is configured as trunk. The primary VLAN is identified as VLAN 100 in Figure 4-9.

Secondary VLANs are configured as isolated or community types. These secondary VLANs are subsets of the primary VLAN. When traffic leaves the private VLAN domain, the traffic that is contained within the secondary VLANs is tagged with the primary VLAN's VLAN ID. This function of private VLANs allows the number of VLANs that are used within a network to be minimized because the traffic is isolated as wanted within the private VLAN domain. However, it presents only as the primary VLAN ID to the rest of the network.

There is only one isolated VLAN that can be configured within the primary VLAN. The isolated VLAN contains isolated ports (I-Ports). Devices within the isolated VLAN cannot communicate directly with any other device within the isolated VLAN or any other secondary VLAN. However, they can communicate with any device within the primary VLAN (promiscuous ports). To communicate with and port other than P-Ports, the I-Ports must communicate through a Layer 3 router that is connected to a P-Port. The isolated VLAN is identified as VLAN 101 in Figure 4-9.

One or more community VLANs are permitted with the primary VLAN. Community VLANs contain community ports (C-Ports). Devices within a community VLAN can communicate with any other device within that community VLAN or the primary VLAN. However, they cannot communicate with any other secondary VLAN. Community VLANs are identified by VLANs 102 and 103 in Figure 4-9 on page 65.

PVLANs can span multiple switches by connecting the switches with PVLAN Trunk ports. A PVLAN Trunk port can be an individual port or a portchannel that is configured as a trunk and carries the primary VLAN and all of the secondary VLANs. In Figure 4-9 on page 65, the PVLAN Trunk carries VLANs 100, 101, 102, and 103.

Table 4-2 lists the permitted traffic flow between the private VLAN ports.

Table 4-2 Permitted traffic flow between the private VLAN ports

	P-Port	**I-Port**	**C1-Port**	**C2-Port**	**PVLAN Trunk**
P-Port	Allow	Allow	Allow	Allow	Allow
I-Port	Allow	Deny	Deny	Deny	Allow
C1-Port	Allow	Deny	Allow	Deny	Allow
C2-Port	Allow	Deny	Deny	Allow	Allow
PVLAN Trunk	Allow	Allow	Allow	Allow	Allow

4.4.1 Why use private VLANs

Private VLANs are commonly used for the following tasks:

- Isolate Layer 2 traffic between ports
- Improve security
- Maximize the use of VLAN IDs
- Simplify IP addressing and subnetting

4.4.2 Full Private VLAN and Private VLAN Edge

Support for Full Private VLAN is added in the TOR switch firmware v7.9 and embedded switch firmware v7.8 and later. Before this version, only Private VLAN Edge was supported.

Private VLAN Edge means that the private VLAN configuration has local significance only to the switch. This configuration directly affects community VLANs that are configured across multiple switches because it must pass through a Layer 3 router before it can communicate community ports on another switch after the traffic departs the switch.

Full Private VLAN means that the switch can pass the private VLAN traffic across all switches that are interconnected with private VLAN trunk ports. A private VLAN trunk port means that the ports that connect the switches must carry the primary VLAN and all of the secondary VLANs.

4.4.3 Private VLANs and STP

When private VLANs are used with STP, the primary and secondary VLANs must be within the same STP group (STG) if the private VLAN is trunked to another switch. If the primary and secondary VLANs are not in the same STG, the STP does not properly block the secondary and primary VLANs.

If the STP mode is PVRST, all of the VLANs must be configured to use Spanning Tree group 1. If MSTP is used, the primary and secondary VLANs must be in the same STG.

4.4.4 Configuring Private VLANs

Private VLAN configuration is a straightforward process. Complete the following steps to implement the private VLAN configuration that is shown in Figure 4-9 on page 65:

1. Enter configuration mode by running the **`configure terminal`** command.
2. Create the Primary and Secondary VLANs and place them in the same STG, as shown in the following commands:

```
vlan 100-103
stg 1
```

3. Assign the Primary VLAN, as shown in the following commands:

```
vlan 100
name P-VLAN
private-vlan primary
```

4. Assign the Isolated VLAN, as shown in the following commands:

```
vlan 101
name I-VLAN
private-vlan isolated
```

5. Assign the Community VLANs, as shown in the following commands:

```
vlan 102
name C1-VLAN
private-vlan community
vlan 102
name C2-VLAN
private-vlan community
```

6. Associate the Secondary VLANs (Isolated and Community) with the Primary VLAN, as shown in the following commands:

```
vlan 100
private-vlan association 101-103
```

7. Assign promiscuous ports, as shown in the following commands:

```
interface port 17-20
switchport mode private-vlan
switchport private-vlan mapping 100
```

8. Assign ports to the secondary VLANs, as shown in the following commands:

```
interface port 21-30
switchport mode private-vlan
switchport private-vlan host-association 100 101
interface port 31-40
switchport mode private-vlan
```

```
switchport private-vlan host-association 100 102
interface port 41-50
switchport mode private-vlan
switchport private-vlan host-association 100 103
```

Note: If more than the private VLAN is to be carried on the port, set the port mode to trunk and allow the wanted VLANs. A port can use private and regular VLANs.

9. If the private VLAN is to be shared with another switch, trunk the primary and secondary VLANs on the interconnection ports, as shown in the following commands:

```
interface port 61-64
switchport mode trunk
switchport trunk allowed vlan add 100-103
```

10. Verify the configuration by running the following commands:

```
show vlan
show interface information
```

4.4.5 Private VLANs and UFP

PVLANs can be used with UFP virtual ports (vPorts) and configured similar to the physical switch ports. The vPorts can be configured as normal and then the PVLAN is assigned by configuring the vPort's *network default-vlan* or, if the mode is trunk, it can be assigned within the PVLAN's VLAN configuration by using the **vmember** command.

Example 4-3 shows a UFP vPort configuration to the isolated VLAN.

Example 4-3 UFP vPort configuration to the isolated VLAN

```
ufp port 51 vport 1
      network mode access
      network default-vlan 101
      qos bandwidth min 50
      enable
      exit
```

4.4.6 Private VLANs and VLAG

PVLANs can be configured on a pair of switches that are configured for VLAG. If you configure PVLANs and VLAG, complete the following tasks:

- Configure the PVLANs on both switches with the same primary and secondary VLANs.
- Configure the ISL as a PVLAN Trunk to carry the primary VLAN and all of the secondary VLANs.
- To minimize the differences between the switches are minimal, configure the same ports on each switch for the PVLANs.

4.4.7 Verifying the Private VLAN

Run the **show vlan** command to verify the PVLAN configuration, as shown in Figure 4-10.

```
VLAN    Name                             Status MGT      Ports
----  --------------------------------  ------ --- ------------------------
1     Default VLAN                       ena    dis  52-60
100   P-VLAN                             ena    dis  1 5 9 13 17-20 61-64
101   I-VLAN                             ena    dis  1 5 9 13 21-30 51 61-64
102   C1-VLAN                            ena    dis  1 5 9 13 31-40 61-64
103   C2-VLAN                            ena    dis  1 5 9 13 41-50 61-64
1000  Shared Data VLAN 1000              ena    dis  1 5 9 13 17-50 61-64
1001  Shared Data VLAN 1001              ena    dis  1 5 9 13 17-50 61-64
4095  Mgmt VLAN                          ena    ena  MGT
Primary  Secondary  Type             Ports                 vPorts
-------  ---------  ---------------  --------------------  ------------------
100      101        isolated         21-30                 51.1
100      102        community        31-40                 empty
100      103        community        41-50                 empty
```

Figure 4-10 Sample show vlan command output

In Figure 4-10 (which matches the sample configuration on a G8264 switch), the following output is shown:

- Ports 1 - 13 are the 40 Gb ports that are to be configured as the VLAG ISL and trunk all of the VLANs including the PVLANs (100 - 103) and any other data VLANs (1000 - 1001).
- Ports 61 - 64 are to be used as uplinks off from the switch and are configured as a portchannel. These ports also trunk all of the VLANs.
- Ports 17 - 20 are configured as P-Ports and can communicate to all of the PVLAN ports. These ports are to be configured as a trunk (switchport mode trunk) and carry the shared data VLANs (1000-1001).
- Ports 21 - 30 are configured as I-Ports and are to be configured as a trunk to carry the shared data VLANs.
- Ports 31 - 30 and 41 - 50 are configured as C-Ports for the respective community VLANs and are configured as a trunk to carry the shared data VLANs.
- Port 51 is configured for UFP with the vPort 1 configured as an I-Port.

Run the **show interface information** command to validate the port's membership in each VLAN and the default VLAN (PVID) and trunk configuration.

4.5 Virtual Fabric Mode and UFP

vNIC is the first virtual NIC technology to be used in the BladeCenter 10Gb Virtual Fabric Switch Module. It is included in the Flex System environment to allow customers that have standardized on vNIC to use it with the Flex System solutions.

UFP is the current direction of Lenovo NIC virtualization, and provides a more feature-rich solution compared to the original vNIC Virtual Fabric mode. As with Virtual Fabric mode vNIC, UFP allows splitting a single 10 Gb port into four virtual NICs (that is, vPorts).

For more information about options for vNIC and UFP, how they work, and preferred practices, see *NIC Virtualization in Flex System Fabric Solutions*, SG24-8223, which is available at this website:

http://lenovopress.com/sg248223

4.6 Layer 2 failover

Layer 2 failover on Lenovo switches works with NIC teaming on the servers to prevent a black hole when all uplinks out of a switch go down. This feature is important in embedded environments, but also applies to stand-alone Lenovo switches. When the uplinks out of an embedded switch go down in an embedded environment, the server-facing ports normally are still up. In this situation, teaming on the servers does not know that the path via the uplinks went down and that it needed to fail over. When failover is enabled on the switches, the switch also shuts down the server facing ports when the uplinks being monitored go down, which alerts NIC teaming that this switch no longer has a path out and failing over.

After the monitored uplinks are restored, failover automatically re-enables the server facing ports, which informs NIC teaming that this path is available again. Failover is triggered not only on the monitored uplinks being in a down state, but also if the monitored uplinks are all in a spanning-tree blocked state.

Failover offers two ways of monitoring and disabling INT ports: Auto Monitoring (AMON), which controls all INT ports at a time, and Manual Monitoring MMON), which can be used to selectively disable certain INT ports. AMON and MMON can be configured to work independently or with other features, such as vNIC, UFP, or SPAR/XPAR.

For more information about options for Layer 2 Failover and preferred practices, see *NIC Virtualization in Flex System Fabric Solutions*, SG24-8223, which is available at this website:

http://lenovopress.com/sg248223

4.7 IGMP Snooping considerations

Internet Group Management Protocol (IGMP) snooping is a feature to control multicast flows in Layer 2 networks. By default, IGMP snooping is disabled on Lenovo switches, and all multicast traffic in an L2 network is flooded as though it is broadcast traffic. To limit multicast flows to devices and ports only that need to see a specific stream, IGMP snooping can be enabled. Consider the following points when you are implementing IGMP snooping on Lenovo switches:

- All Lenovo switches support IGMP V1, V2, and V3.
- Some other features might limit the use of IGMP snooping. It is important to check the Application Guide for the product and feature to ensure IGMP use is not restricted. For example, consider the following points:
  - IGMP V3 snooping currently is not supported on vLAG enabled switches; however, such support is planned for a coming release for the G8264 (check the release notes and Application Guides for the latest feature support list for a product). IGMP V3 snooping currently is not supported on stacked switches.

– The number of supported multicast group entries on a Lenovo switch can vary, depending on the following factors:

  - Version of code
  - Switch model
  - Features enabled
  - Switch profile that is configured

  For example, an EN4093 switch supports up to 3072 IGMP entries in version 7.8 code; however, if stacking is enabled, the number of supported entries is reduced to 1022. For more information, see the Application Guide for the specific product and version of code.

- If an environment includes multicast routers, they can be used to perform keep-alive queries and keep multicast join requests from expiring. If you are working in an environment where only local multicast is being used (no mrouters), the IGMP querier feature of Lenovo switches can be enabled to provide this keep-alive query service that is normally provided by a mrouter.

4.8 Link aggregation

This section provides information about link aggregation. Lenovo switches support static (PortChannel) and dynamic (LACP) link aggregation modes. The key difference between these modes is that static aggregation is unconditional and always in effect on ports where it is configured. Dynamic aggregation uses an interactive protocol between both devices, which helps protect against cabling errors and other errors that can cause unwanted effects.

In general, dynamic aggregation with LACP is preferred. It is standards-based and supported by all network equipment vendors. Some server operating systems and utilities from NIC vendors also support LACP. There are some conditions where static link aggregation should be considered when the server operating system (or other software) that is being connected to does not support LACP.

Note: The commands on our current firmware that are used to configure static or dynamic aggregation groups are different from each other. For more information, see the Command Reference manuals and the following sections.

Static PortChannel

This section describes Static PortChannel, which has the following form:

```
Static PortChannel: PortChannel <n> port <list or range of port numbers> [enable]
```

Identifiers for static PortChannels range 1 - 64, except on the Flex System Interconnect Fabric, where they range up to 288.

Ports in the same channel must have the same attributes in the following areas or the channel does not form successfully:

- VLAN membership, including native VLAN
- Spanning Tree options
- Bandwidth (ports with different bandwidths cannot be channeled together)

LACP (dynamic) PortChannel

This section describes LACP (dynamic) PortChannel.

LACP channels are configured primarily on lists or ranges of ports, where the LACP key and state are configured, and involves the following commands:

```
LACP key <1-65535>
LACP mode {active|passive|off}
```

The LACP key number must be the same on all ports to be put in a common aggregation. Mode **active** often is the preferred mode when enabling LACP.

Aggregator numbers and trunk numbers

Two numbers are used to identify aggregation groups. A *trunk number* is used in the **Port-Channel** command when a static aggregation is configured, or with the static binding option for an LACP aggregation. This number is equivalent to a Cisco Port-Channel number, and is used similarly in operator commands.

Static aggregations have trunk numbers 1 - 64; trunk numbers 65 and greater are used for LACP aggregations and are assigned automatically.

Aggregator numbers are also assigned automatically and are typically learned by running the **show lacp info** command. These numbers are used in other **show lacp** commands. Figure 4-11 shows an example of this command and its output. The aggregator number often is (but not always) equal to the sequential port number of the lowest port that is included in the aggregation group.

```
show lacp info ?
  state  Show lacp information of interfaces matching a state
  <cr>
  |  Search output for strings

bay-1#show lacp info state up
port    mode    adminkey  operkey   selected   prio  aggr  trunk  status  minlinks
--------------------------------------------------------------------------------
EXT1    active       102      102    yes      32768    43     67    up        1
EXT2    active       102      102    yes      32768    43     67    up        1
EXT3    active       304      304    yes      32768    45     65    up        1
EXT4    active       304      304    yes      32768    45     65    up        1
EXT9    active      5152     5152    yes      32768    51     66    up        1
EXT10   active      5152     5152    yes      32768    51     66    up        1
(*) LACP PortChannel is statically bound to the admin key
```

Figure 4-11 Output of the show lacp info command

Default LACP key values

Every port on Lenovo switches has a pre-assigned LACP key value that is equal to its port number, except for the Flex System Interconnect Fabric, which uses a slightly different scheme.

For example, a G8264 switch pre-assigns values 1 - 64. For this reason, avoid low numbered values when you are assigning LACP keys.

4.8.1 Trunk hashing configuration

A set of aggregated links does not completely match the performance characteristics of a single link with the same total bandwidth. This issue occurs because all link aggregation techniques use an algorithm to determine which physical link is to be used to transmit a packet. These algorithms are referred to as *hashing* algorithms. When two devices are connected by a set of aggregated links, each side uses a hashing algorithm and each side can configure that algorithm independently from the other side.

The purpose of these options is to allow the choice of an option that best provides even allocation of traffic across the available aggregated links. The reason that aggregated links might not deliver the same performance as a single link is that the traffic can be unevenly allocated across the available links, which saturates some of them and leaving others nearly idle.

Lenovo switches have the following options for configuration of the hashing algorithm, which is all options for the **`portchannel thash`** command or the **`portchannel hash`** command, depending on switch model and firmware (some switch models have different option, so it is always good to check the Application Guide, release notes, and the output of the **`portchannel ?`** command for current options on your switch):

- L3 hashing, which operates by default if a packet contains IP addresses. L3 hashing uses the IP addresses to choose a physical link and it can be configured to consider the source IP address, destination, or both. The default option for L3 is to use source and destination IP address. The command to use L2 hashing for packets that contain IP addresses is **`portchannel thash l3thash l3-use-l2-hash`**, which uses the configured L2 hash options for IP traffic.
- L2 hashing, which by default is used for packets that do not have IP addresses. L2 hashing uses MAC addresses to choose a physical port, and can be configured to use source MAC addresses, destination, or both. The default for L2 hashing is to use both source and destination MAC addresses.
- Ingress port hash hashing, which uses the ingress port of the packet to determine which aggregated port to use for the outbound aggregation. This option can be used with the L2 options, L3 options, or both. It is disabled by default.
- L4port hashing, which uses the TCP or UDP destination port to determine which physical link to use. This option can be used in with the L3 options and it is disabled by default.
- FCoE hashing, which applied to FCoE traffic only that is sent over an Ethernet link aggregation group and that offers several FCoE specific parameters. L2 hashing for FCoE likely is to be limited because FCoE traffic typically flows to and from the Fibre Channel Forwarder (FCF).

Measuring traffic on parallel links

To choose the appropriate option, a process or trial-and-error often is used unless good information about traffic characteristics is available. This process includes making a best guess initial choice of options and measuring the traffic flowing over the parallel links to determine whether it is imbalanced to the point that another option is better.

The following commands can be used to gather this information:

- **`show interface port <ID | list | range> bitrate-usage`** provides repeated displays of the current usage of the selected ports in each direction.
- **`show interface port <ID | list | range> interface-rate`** provides a one-time display of octets in and out and other measures.

- **`show interface port <ID | list | range> interface-counters`** can be used after clearing the counters to gather statistics over a longer interval of time.

Changing the parameters on a switch influences only the packets that are egressing from that switch. The device at the other end, which might be from a different vendor, needs a similar configuration to ensure that traffic coming into a Lenovo switch is well-balanced across a link aggregation group.

4.8.2 Options for LACP configuration

This section describes the options for LACP configuration.

LACP timers

The LACP timeout option can be set to long (30 seconds, which is the default setting), or short (1 second). Most vendors also default to long, but a few (for example, Juniper) default to short. For proper operation, both sides must agree on the same LACP timers. Based on several factors, it is best to use long timers whenever possible. The use of short timers on both sides can affect switch stability, and result in false positives for LACP failure, which leads to less than stable operation. This parameter is set once per switch, or per stack and fabric as appropriate.

Suspend-individual

This option on a *port-channel* command (or on the *lacp* command starting with 7.9 firmware, where it is the default setting), which configures an LACP PortChannel so that if an individual port does not receive LACPDU packets from its counterpart on the other device, the port goes into a suspended state and not pass any traffic. In general, this result is desirable; if it is not configured, after a period the port functions as though it were an individual port and not part of any aggregation.

The primary case when this option is not helpful is on server-facing ports where the server uses a form of network boot (PXE, BOOTP, and so on). Because the code that provides the LACP protocol is typically not available before boot, the port does not pass any traffic and the network boot fails.

Static binding

This option prevents misconfigurations from causing two parallel LACP aggregations to form between two devices. For example, if one device has four ports configured with key 100, and the adjacent device has two ports configured with 100 and two ports with 200, and these ports are all linked, two 2-port aggregations might be formed. This configuration might create an immediate network loop if STP is not configured, and such a loop is undesirable. In general, use static binding unless you want the ability to form multiple parallel link aggregations.

Note: For firmware before 7.9, the **`suspend-individual`** option is required when a static binding is configured, as shown in the following command syntax:

```
portchannel <x> lacp key <y> [suspend-individual]
```

LACP configuration

The use of a systematic scheme for numbering LACP keys can be useful for debugging.

All ports on Lenovo switches have a default configuration with a key that is equal to their sequence number on the switch and the LACP mode set to off. Therefore, low-numbered keys can cause problems. The default numbering that is based on the port sequence number also applies to stacked configurations, where there can be a total of hundreds of ports.

The following possible numbering scheme uses the port numbers in the keys:

- If ports 10 and 11 are bound together, use key 1011.
- If port 9 is set up alone with LACP, such as for use in vLAG, use key 1009. In this case, assume that the other switch also uses port 9 as part of the same aggregation.
- If ports 10-12 are bound together, use port 1012; if more than two ports are bound together (such as ports 20-22), key 2022 can be used.

Because LACP keys have only local significance, the device at the other end of the links can use a key appropriate for the ports that are used on its side.

Sample LACP configurations

Figure 4-12 shows the general configuration for using LACP. It is not at all similar to the configuration for a static link aggregation and it does not include the **portchannel** command.

```
interface port [range | list]
lacp key <1-65535>
lacp mode active
```

Figure 4-12 Typical LACP configuration

Although *passive* mode is also supported, there is no advantage in using it. If both sides of a link aggregation use *active* mode, the side that starts the protocol first takes the active role. If both sides use the **passive** option, the LACP protocol does not complete and the links do not pass traffic.

The following commands can be used for most configurations. The command that is shown in Figure 4-13 is supported starting in firmware version 7.9.

```
port-channel <65-128> lacp key <1-65535>
port-channel <65-128> lacp key <1-65535> suspend-individual
```

Figure 4-13 Static binding and suspend options

The command that is shown in Figure 4-14 is supported in previous versions and it is recommended that it not be used on server-facing ports where PXE, BOOTP, and similar protocols are used. (Suspend-individual is the default in firmware 7.9).

```
interface port <ID | range | list>
no lacp suspend-individual
```

Figure 4-14 Suspend option for firmware 7.9; for use where PXE or similar protocols are used

LACP without static binding: Multiple aggregation example

This topology often is caused by a configuration or cabling error. If multiple parallel connections are intended, different LACP keys can be used on both devices. If STP is not enabled in this scenario, a network loop and broadcast storm occur. Ports 1 - 4 of each switch are connected to the same port numbers on its partner and the configurations are deliberately set, as shown in Figure 4-15 and Figure 4-16.

Figure 4-15 shows the bottom switch configuration.

```
int port 1-2
lacp key 1200
lacp mode act
lacp key 3-4
lacp key 3400
lacp mode act
```

Figure 4-15 Bottom switch configuration

Figure 4-16 shows the top switch configuration.

```
int port 1-4
lacp key 1400
lacp mode act
```

Figure 4-16 Top switch configuration

The result is two distinct two-port link aggregations, as shown in the **`show lacp info`** output. Figure 4-17 shows the output from the bottom switch.

```
G8264CS-Bottom(config-if)#sho lacp inf state up
port    mode    adminkey  operkey   selected   prio  aggr  trunk  status  minlinks
--------------------------------------------------------------------------------
1       active      1200     1200    yes      32768     1     67    up        1
2       active      1200     1200    yes      32768     1     67    up        1
3       active      3400     3400    yes      32768     3     68    up        1
4       active      3400     3400    yes      32768     3     68    up        1
```

Figure 4-17 Output from the bottom switch with two aggregations

Figure 4-18 shows the output from the top switch.

```
8264CS-Top(config-if)#show lacp inf state up
port    mode    adminkey  operkey   selected   prio  aggr  trunk  status  minlinks
--------------------------------------------------------------------------------
1       active      1400     1400    yes      32768     1     65    up        1
2       active      1400     1400    yes      32768     1     65    up        1
3       active      1400     1400    yes      32768     2     67    up        1
4       active      1400     1400    yes      32768     2     67    up        1
```

Figure 4-18 Output from top switch with the same key but still two aggregations

If a static trunk binding is configured on the top switch, only one aggregation is formed, as shown in Figure 4-19.

```
8264CS-Top(config)#portchannel 65 lacp key 1400 suspend-individual
Warning: This may restart LACP on ports with admin key 1400.
Proceed? (y/n) ? y
8264CS-Top(config)#
Feb 16  1:39:30 8264CS-Top NOTICE  lacp: LACP is down on port 1
Feb 16  1:39:30 8264CS-Top NOTICE  lacp: LACP is down on port 2
Feb 16  1:39:30 8264CS-Top NOTICE  lacp: LACP is down on port 3
Feb 16  1:39:30 8264CS-Top NOTICE  lacp: LACP is down on port 4

Feb 16  1:39:31 8264CS-Top NOTICE  lacp: LACP misconfiguration detected on port 3: Port
partner key:3400, does not match PortChannel partner key:1200
Feb 16  1:39:31 8264CS-Top NOTICE  lacp: LACP is suspended on port 3

Feb 16  1:39:31 8264CS-Top NOTICE  lacp: LACP misconfiguration detected on port 4: Port
partner key:3400, does not match PortChannel partner key:1200
Feb 16  1:39:31 8264CS-Top NOTICE  lacp: LACP is suspended on port 4

Feb 16  1:39:31 8264CS-Top NOTICE  lacp: LACP is up on port 1
Feb 16  1:39:31 8264CS-Top NOTICE  lacp: LACP is up on port 2
sho lacp inf
port    mode    adminkey  operkey   selected   prio  aggr  trunk  status  minlinks
--------------------------------------------------------------------------------
1       active      1400     1400    yes      32768     1     65*   up        1
2       active      1400     1400    yes      32768     1     65*   up        1
3       active      1400     1400  suspended  32768    --     65*  down       1
4       active      1400     1400  suspended  32768    --     65*  down       1
```

Figure 4-19 Result of static LACP binding

If suspend-individual is disabled on a port, the port goes into individual mode, as shown in Figure 4-20.

```
Aug 20 20:26:37 G8264CS-Bottom NOTICE  lacp: LACP is down on port 1

Aug 20 20:26:47 G8264CS-Bottom NOTICE  lacp: LACP port 1 is individual for not receiving
any LACPDUs
sho lacp inf
port    mode    adminkey  operkey   selected   prio  aggr  trunk  status  minlinks
--------------------------------------------------------------------------------
1       active      1200     1200  individual 32768    --     --   down       1
2       active      1200     1200    yes      32768     1     67    up        1
3       active      3400     3400    yes      32768     3     68    up        1
4       active      3400     3400    yes      32768     3     68    up        1
```

Figure 4-20 Port in individual mode

At this point, there are three parallel connections between the two switches: port 1 in individual mode and two parallel LACP aggregations. Two of these paths are blocked because STP is enabled.

If the individual port is reconfigured to be suspended, only the LACP links stay active, as shown in Figure 4-21.

```
G8264CS-Bottom(config-if)#int port 1
G8264CS-Bottom(config-if)#lacp suspend-individual
This command will be applied to ports with the same key.

Aug 20 20:30:30 G8264CS-Bottom NOTICE  lacp: LACP port 1 is suspended for not receiving
any LACPDUs

sho lacp inf
port    mode    adminkey  operkey   selected   prio  aggr  trunk  status  minlinks
--------------------------------------------------------------------------------
1       active      1200     1200  suspended  32768    --     --   down       1
2       active      1200     1200    yes      32768     1     67    up         1
3       active      3400     3400    yes      32768     3     68    up         1
4       active      3400     3400    yes      32768     3     68    up         1
```

Figure 4-21 Port is suspended

Do not use the suspend-individual option for server-facing ports, especially where network boot protocols, such as PXE and BOOTP are deployed.

4.9 Spanning Tree Protocol

STP is a commonly used protocol to actively block network loops. The consequences of a network loop are dramatic and usually result in significant network outages. The use of STP can protect your network from these potential outages by providing automatic activation of redundant paths.

Although STP is one of the most misunderstood and problematic configuration elements in networking, successful implementation can be achieved if you are careful and follow a few simple rules that are described in this section. STP is an active protocol, which means that transmissions that occur on the wire and missteps in the configuration can result in inefficient network paths and potential network outages. Therefore, STP configuration changes should be conducted during scheduled network downtimes.

4.9.1 STP fundamentals

STP was originally defined in the IEEE 802.1D standard, which defined a single instance of STP. This standard describes how multiple Layer 2 bridges (typically switches) can interact to define a spanning tree blocking any redundant paths between network nodes.

When you configure STP, you must consider the hierarchy of the spanning tree. The tree has a root bridge with a subtree below it. Then, each subtree has a designated root bridge that points back to the root bridge. The subtrees fan out all the way down to the edge devices.

Figure 4-22 shows an example network.

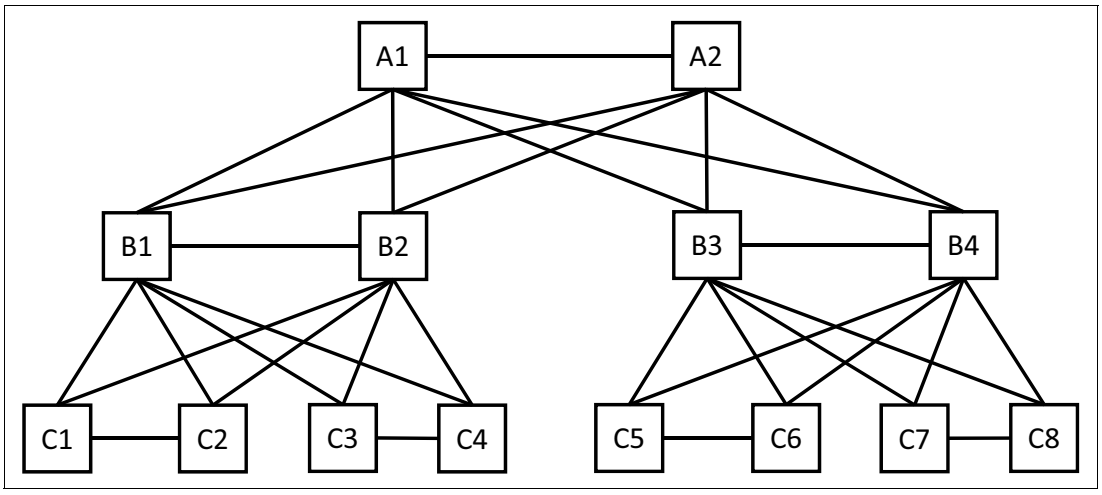


Figure 4-22 Example network

As shown in Figure 4-22, multiple loops are in the network and, if nothing is done to block the loops, a single broadcast packet (such as an ARP) can rapidly overrun the network. This issue occurs because each broadcast packet replicates two or more times while it is being flooded by each switch. STP blocks the redundant links to prevent the network loops through a negotiation. This negotiation uses a packet called a Bridge Protocol Data Unit (BPDU) that is transmitted by the switches.

To configure STP, you must first determine which switch should be the root bridge, which often is the core of the network. In the example, this switch is A1 or A2. Then, you must determine each level of tree down to the edge (in this example, C1-C8).

After you determine the hierarchy of the tree, you must determine the bridge priority to use at each level that is used for STP to determine the root of the tree. Bridge priorities have a value of 0 - 61,440 incremented in intervals of 4096 where a switch with the lowest bridge priority is selected as the root bridge. If the bridge priority is equal, the switch with the lower MAC address is the root.

If you want a switch to be the root bridge, set the bridge priority to 0 but exercise caution when this setting is used in a network. By default, most switches set their default bridge priority to 32768, which includes the RackSwitch TOR switches. The Flex System embedded switches use 61440 because these switches are normally installed as edge devices.

STP then uses the bridge priority to determine the designated root bridge for each loop that is the preferred path to the root bridge or the tree's root bridge. The switch that is determined to be the furthest path from the root bridge blocks ports by placing them to discarding to eliminate the network loops. Figure 4-23 shows an example network with bridge priorities.

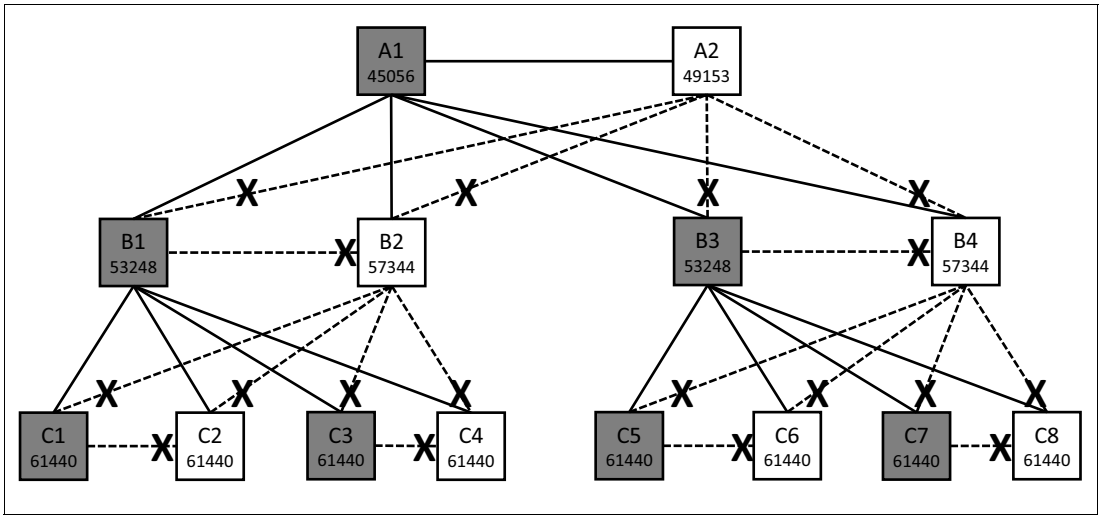


Figure 4-23 Example network with bridge priorities

Figure 4-23 displays the example network with some example bridge priorities configured, assuming that the subtree is inserted at the edge of the network. Devices C1 - C8 are edge devices; therefore, the bridge priority should be set to 61440. The next layers should have a primary and secondary bridge configured so each pair should be set to adjacent bridge priorities. B1 - B4 are configured with bridge priorities 53248 and 57344 and switches A1 and A2 use the bridge priorities of 45056 and 49153. These values are deterministic on the target network.

If the preceding bridge priorities are used, each level of the tree determines the designated bridge for each level (indicated by the gray box in Figure 4-23), which has the preferred path to the root bridge. The non-preferred paths are blocked (designated by X in Figure 4-23) on the switch that is furthest from the root bridge. The alternative paths are marked by dashed lines in Figure 4-23 and all loops are blocked.

Although there are many other elements to STP, the bridge priorities are the most important element to understand because this element is the primary element that is used to form the hierarchy of the spanning tree. When any link on a port changes, communication between devices in the spanning tree might be interrupted while STP renegotiates the best path to the root bridge, which is called *STP convergence*. This concept is important because if you reboot a device on the LAN, network traffic can be interrupted.

The STP standard evolved as the Layer 2 protocols evolved to include support for protecting loops in multiple VLANs. One such standard developed by Cisco is called Per VLAN Spanning Tree (PVST) and Per VLAN Spanning Tree Plus (PVST+) where a spanning tree instance is created for each VLAN. Each STP instance uses a BPDU that is tagged with the VLAN ID to negotiate the STP protocol. PVST/PVST+ STP instance 1 uses an untagged BPDU, which is compatible with the IEEE standard STP. STP instance 1 is the only instance that can control more than one VLAN because all of the other instances use the unique VLAN ID to tag the BPDU.

Rapid STP (RSTP) defined by IEEE 802.1w evolved from STP to significantly reduce the amount of time it takes for a network to converge from 40 to 50 seconds down to 6 seconds or less. RSTP is compatible with an earlier version with STP and is incorporated into the IEEE 802.1D-2004 standard. Cisco extended PVST/PVST+ to include RSTP extensions with Rapid Per VLAN Spanning Tree (RPVST) or Per VLAN Rapid Spanning Tree (PVRST).

Finally, the IEEE community added Multiple Spanning Tree Protocol (MSTP), which was originally defined in the IEEE 802.1s standard and later merged into IEEE 802.1Q-2005. MSTP is similar to RPVST in that it contains multiple STP instances. However, unlike RPVST, MSTP allows multiple VLANs to be defined in all of the STP instances. MST also limits the number of STP instances to 33 with IDs 0 - 32 and maps these into a single BPDU to provide less processing overhead than RPVST, which uses a BPDU for each VLAN. MSTP is fully compatible with MSTP with the Common Instance Spanning Tree (CIST), which is instance 0.

In summary, the following major implementations of STP are available:

- Rapid Spanning Tree Protocol (RSTP)
- Rapid Per VLAN Spanning Tree (RPVST)
- Multiple Spanning Tree Protocol (MSTP)

Another STP mode to consider is to disable STP. The way switches function with STP disabled is not defined in IEEE 802.1D-2004, so care should be taken to understand how the device works in this mode. The following section describes how this mode works in the Lenovo switches.

Each of these modes is different and it is important to select the proper STP mode (type) when integrating into a network. Network instability and outages occur if the same STP mode is not used on every network device.

4.9.2 How STP is implemented on the Lenovo switches

The Lenovo switches implement four STP modes that follow the STP protocols that are described in 4.9.1, “STP fundamentals” on page 78. The following sections describe how the modes are implemented by Lenovo.

Per VLAN Rapid Spanning Tree

PVRST is the default STP mode in the switch. The following command is used to enable PVRST:

```
spanning-tree mode pvrst
```

In PVRST mode, there are 128 STP groups (or instances) with 127 available to assign to VLANs. Unlike Cisco when the VLAN ID identifies the STP instance, the STP group is a value of 1 - 128 with Lenovo NOS where the VLANs are assigned to the STP group.

VLANs are automatically assigned to the group number when the VLAN is created and (when possible) the STP group is automatically selected to be the same as the VLAN ID if the ID is less than 128 and the group is not previously used. STP group 128 is reserved for the management VLAN (4095). STP group 1 functions as you expect with Cisco RPVST assigning more than one VLAN to it and it is compatible with single instance RSTP. If the number of VLANs exceed 128, any other VLANs are assigned into group 1.

One difficulty with the approach of mapping VLANs to STP group IDs is tracking the ID to which the VLANs are mapped. This task can be easily accomplished by viewing the switch configuration by using the following command:

```
show running-config | include "span"
```

You can also view the switch's STP operational state by using the following command:

```
show spanning-tree | include "Group|VLAN"
```

The VLAN to STP Group assignment is implemented by using the following command:

```
spanning-tree stp GROUP vlan VLAN
```

The following VLAN configuration also can be used:

```
vlan VLAN
stg GROUP
exit
```

In this example, **`VLAN`** is the VLAN ID that is configured and **`GROUP`** is the STP group (1 - 128).

The STP group's bridge priority is configured by using the command:

```
spanning-tree stp GROUP bridge priority PRIORITY
```

In this example, **`GROUP`** is the STP group and **`PRIORITY`** is the STP priority (0 - 61440). The priority is automatically adjusted to the next lowest interval of 4096, as described in 4.9.1, "STP fundamentals" on page 78. For example, if 61000 is selected, the priority automatically is adjusted to 57344 (61440 - 4096).

Multiple Spanning Tree

MSTP can be enabled by using the following command:

```
spanning-tree mode mst
```

Before MSTP mode can be selected, the name and revision number must be configured with the same value as all other devices in the network. The name is a character string of up to 32 case-sensitive characters and the revision is a value of 0 - 65535 where the default value is 1. Most vendors set the default revision number as 1, except for Cisco.

The following command is used to configure the name and revision:

```
spanning-tree mst configuration
name NAME_STRING
revision REVISION_NUMBER
exit
```

The MSTP configuration can be validated by using the command **`show spanning-tree mst`** configuration.

For MSTP to properly balance the trees, the VLANs must be mapped to the same instances throughout the network. The assignment is done by using the following command:

```
spanning-tree mst configuration
instance INSTANCE vlan VLANS
exit
```

In this example, **INSTANCE** is the MSTP instance 0 - 32 and **VLANS** is the range of VLANS to be assigned. The VLAN mappings can be validated by comparing the MSTP Digest, which is the common calculated value defined by the MSTP standard and used by all devices in the tree. The MSTP Digest can be displayed by using the following command:

```
show spanning-tree | include Digest
```

The use of this command results in the following value:

```
Mstp Digest: 0x87957342f6b0029d887baaec6212b0bf
```

If the MSTP Digest is the same on all devices, the VLAN mappings are the same on all devices.

The MSTP instance bridge priorities function the same way as in all of the STP standards and can be set by using the following command:

```
spanning-tree mst INSTANCE priority PRIORITY
```

Rapid Spanning Tree

Rapid Spanning Tree is configured by using the following command:

```
spanning-tree mode rstp
```

This configuration assigns all of the VLANs into the single RSTP instance. RSTP is also fully compatible with the non-rapid STP.

Disable Spanning Tree

Spanning Tree can be globally disabled by using the following command:

```
spanning-tree mode disable
```

Globally disabling STP disables all local STP processing on the switch and any BPDU that is received is forwarded as is any other L2 packet, which allows the upstream switches to process all STP control.

4.9.3 Loop Guard

Loop Guard adds protection against network loops that are caused by improperly functioning remote devices with conditions, such as a unidirectional link failure. This feature monitors the BPDUs that are normally received on STP enabled ports and places the port into a loop-inconsistent blocking state if BPDUs are no longer received. After BPDUs are received again, the port is placed back into a normal error free STP state.

The following command is used to globally enable loop guard:

```
spanning-tree loopguard
```

Loop guard must be enabled on each port where the feature is required by using the command:

```
interface port PORTS
spanning-tree guard loop
exit
```

4.9.4 Lenovo port-specific Spanning Tree 0ptions

Some STP controls are enabled on a per-port basis to enable tighter control of STP. This section describes these port specific features. All of these commands are under the interface port PORTS configuration. PORTS is a range of ports to be configured.

Port fast

The port fast feature enables a port to immediately go into forwarding state. This feature should be enabled on all server-facing ports. If this feature is not enabled, a server that is rebooted can cause a network convergence event that disrupts network traffic and results in lost data. To enable port fast, use the **`portfast`** command.

This command also is referred to as *spanning-tree edge* in some early releases of the firmware. Port fast functions differently in the Lenovo switches than Cisco because if this feature is enabled on a port and a BPDU is received, the port enters the normal STP port processing and blocks if a loop is detected. Cisco places the port in an error disabled state.

Another reason for enabling port fast on client- and server-facing ports is that the ports are immediately placed into forwarding so that traffic is passed. This fact is important for devices that use DHCP for IP configuration because ports time out before receiving proper configuration.

Disabling STP on ports

It can be necessary to disable STP on a per port basis. Unlike globally disabling STP where the BPDUs are forwarded as L2 packets, disabling on a per port basis causes any BPDUs that are received to be discarded. This process can also be referred to as *BPDU filtering*. To disable STP on a per port basis, use the following command for PVRST and RSTP:

```
no spanning-tree stp INSTANCE enable
```

For MST, use the following command:

```
no spanning-tree mst INSTANCE enable
```

As shown in this example, **`INSTANCE`** is the STP instance that is disabled. Each STP instance (group) must be disabled individually.

BPDU guard

BPDU guard error disables a port and generates a log message if a BPDU is received on it. This feature is useful to protect a port against unexpected switches that are plugged in or are from other improper configurations that can result in a loop. It can also be used on L3 only ports where STP is not expected. To enable BPDU guard, use the following command:

```
bpdu-guard
```

Root guard

Root guard protection is used to prevent the STP root bridge for any tree from being learned on a port. If a root bridge is learned, the port is error disabled and a log message is generated. To enable root guard, use the following command:

```
spanning-tree guard root
```

4.9.5 Changing STP standards obsoletes some functions

There were enabled features in STP to allow for fast convergence, such as backbone fast and uplink fast. These features are not supported on the Lenovo switches because they are no longer required because the functions they provide are built into the RSTP protocol, which in turn are included in PVRST and MST.

4.10 Storm Control considerations

Storm Control is a feature that is available on all Lenovo switches. It limits the number of broadcast, multicast, and unknown unicast packets that are allowed into a port. These types of traffic are traditionally flooded (multicast is not flooded if IGMP snooping is enabled and unknown multicast flooding is disabled, but that setting is not the default setting), and thus are the kinds of packets that are caught in network loops (storms) and overwhelm the network interfaces and hosts receiving them.

To prevent excessive amounts of this type of traffic from causing issues in an environment, each can be independently throttled to limit the number of packets per second.

Storm control commands are run on the interface by using the following format on all newer codes (except the G8124, which uses a threshold rate in megabits per second as described in the Application Guide):

```
storm-control <broadcast | multicast | unicast> level pps <packets per second>
```

On some older codes, the syntax is as shown in Example 4-4.

Example 4-4 Storm control syntax

```
broadcast-threshold <packets per second>
multicast-threshold <packets per second>
dest-lookup-threshold <packets per second>
```

Configuring Storm Control can be as much art as science. Every environment can have different levels of these types of traffic that are considered normal. Arbitrarily throttling these types of traffic can cause more issues than it might help. The following examples describe different approaches for selecting appropriate values for Storm Control settings:

Important: Do not be too aggressive when you are setting Storm Control values because the switch can discard normal wanted packets of these types (for example, ARPs are a broadcast and you might inadvertently affect the normal ARP process if the broadcast threshold is set too low).

- Monitor the amount of these types of traffic by using the **`show int port X interface-counters`** CLI command and select values above the normal level for the type of traffic that is being filtered (broadcast and multicast can be seen here, but not unknown unicast traffic).
- Use the **`show int port X interface-rate`** command (it displays a report for the utilization for a 1-second period), monitor the broadcast and multicast for some period, and then take the average number and add one or two zeros. For example, if the broadcast averaged 20 or 30 per second, set the value for 2000 or 3000.

- A more scientific (but potentially time-consuming) method is to use a monitoring tool that gathers data over a period and use the averages as reported by this tool to select appropriate values.
- The simplest and least time-consuming way is to pick a number that is fairly safe or high for most environments and set for this value; for example, 2000 packets per second (pps) of broadcast is typically not seen in most production environments.

> **Important:** After Storm Control values are set, it is important to monitor the policy discards on the port (by using the **`show int port X interface-counters`** command). If there are excessive policy discards in normal conditions (which indicates that storm control is being triggered), consider tuning the values higher until policy discards are minimal to none in normal operating conditions.

- When storm control is applied, it controls *only* packets that are coming into a port, not packets that are going out of a port.
- The **`<packets per second>`** setting range can vary from switch model to switch model. For more information about the range that is available for a specific switch, see the Application Guide or Command reference for the specific model and version of code in use. If you have access to a switch, you can use the question mark option to see the supported range.
- Storm control settings for Lenovo switches are different from Cisco switches in that Cisco switches often set this value for a percentage of the total bandwidth. Lenovo switches (except the G8124 switch) set this value to a packet-per-second value.

4.11 Switch Partition

Switch Partitions (SPARs) allow for the creation of multiple partitions within a switch to form up to eight independent switching contexts regarding the data plane of the switch. The traffic within each SPAR is not shared outside of the SPAR instance. Each SPAR contains a unique pool uplink and data ports. The SPAR feature is only available on the Flex System EN4093, EN4093R, CN4093, and SI4093 switches.

SPAR has the following switching modes that it can be configured to use:

- Pass-through domain mode (also known as *transparent mode*), which is the default SPAR mode

  This mode uses a Q-in-Q function to encapsulate all traffic that is passing through the switch in a second layer of VLAN tagging. This mode is the default mode when SPAR is enabled and is VLAN independent owing to this Q-in-Q operation. It passes tagged and untagged packets through the SPAR session without looking at or interfering with any customer assigned tag.

  SPAR pass-through mode supports passing FCoE packets to an upstream FCF, but without FIP Snooping within the SPAR group in pass-through domain mode.

- Local domain mode

  This mode is not VLAN independent and requires a user to create any required VLANs in the SPAR group. Currently, there is a limit of 256 VLANs in Local domain mode.

  Support is available for FIP Snooping on FCoE sessions in this mode. Unlike pass-through domain mode, this mode provides strict control of end host VLAN isolation.

SPAR can be a useful solution in environments where simplicity is paramount.

If a SPAR-like solution is required that uses any of the excluded features that are described in the next section or on a switch other than the EN4093, CN4093 or SI4093, use PVLAN Community VLANs to provide the L2 isolation as described in 4.4, “Private VLANs” on page 65. PVLANs are a standard protocol that can be conveniently extended across multiple switches by using PVLAN trunks.

4.11.1 SPAR restrictions

When you are using SPAR, consider the following points:

- SPAR is disabled by default on the EN4093R and CN4093.
- SPAR is enabled by default on SI4093, with all base licensed internal and external ports, defaulting to a single pass-through SPAR group. This default SI4093 configuration can be changed.
- Port can be a member of only a single SPAR group at one time.
- Only a single uplink path is permissible per SPAR group (can be a single link, a single static aggregation, or a single LACP aggregation). This SPAR enforced restriction ensures that no network loops are possible with ports in a SPAR group.
- STP is turned off on all SPAR ports.
- SPAR cannot be used with UFP or Virtual Fabric vNIC as of this writing. Switch Independent Mode vNIC is supported by SPAR.
- Up to eight SPAR groups per I/O module are supported. This number might be increased in a future release.
- SPAR is not supported by vLAG, stacking, or tagpvid-ingress features.
- Only 32 VLANs are allowed per SPAR

4.11.2 Configuring SPAR

To configure SPAR, complete the following steps:

1. Enter configuration mode by using the **`configure terminal`** command.
2. Select the SPAR instance 1 - 8 by using the **`spar SPAR_ID`** command.
3. Select and uplink port, static, or dynamic portchannel (only a single port or portchannel can be configured as an uplink to prevent network loops) by using the **`uplink adminkey|port|portchannel UPLINK`** command.
4. Configure the mode (default is passthrough) by using the **`domain mode local|passthrough`** command.
5. Configure the default SPAR client ports by using the **`domain default member PORT_RANGE`** command.
6. Configure the VLANs.

   For pass-through SPARs, it is not necessary to configure the VLANs, the default tunnel VLAN for SPARs 1 - 8 are 4081 - 4088. To optionally override the default VLAN for pass-through, you must configure the default VLAN for local mode SPARs by using the **`domain default vlan VLAN_ID`** command.

7. For local domain mode, configure each non-default VLANs up to 32 VLAN instances (VLAN_INSTANCE = 1 - 32) by using the following commands:

```
domain local VLAN_INSTANCE vlan VLAN_ID
domain local VLAN_INSTANCE member PORT_RANGE
domain local VLAN_INSTANCE enable
```

8. Enable the SPAR by using the **`enable`** command.
9. Validate the SPAR configuration by using the **`show spar SPAR_ID`** command.

4.12 BootP and DHCP relay

BootP and DHCP relay act as a relay agent for clients that request a DHCP address; the DHCP server is on a different IP subnet. Acting as a relay agent, the switch can forward a client's IPv4 address request to up to five BOOTP and DHCP servers. In addition to the five global BOOTP and DHCP servers, up to five domain-specific BOOTP and DHCP servers can be configured for each of up to 10 VLANs.

When a switch receives a BOOTP and DHCP request from a client that is seeking an IPv4 address, the switch acts as a proxy for the client. The request is forwarded as a UDP unicast MAC layer message to the BOOTP and DHCP servers that are configured for the client's VLAN, or to the global BOOTP and DHCP servers if no domain-specific BOOTP and DHCP servers are configured for the client's VLAN. The servers respond to the switch with a Unicast reply that contains the IPv4 default gateway and the IPv4 address for the client. The switch forwards this reply back to the client.

4.12.1 Layer 3 single switch

The topology and configuration example that is described in this section uses a G8264 switch with two Layer3 interfaces and BootP configured to forward to a DHCP server.

Figure 4-24 shows a BootP and DHCP request process to a single switch.

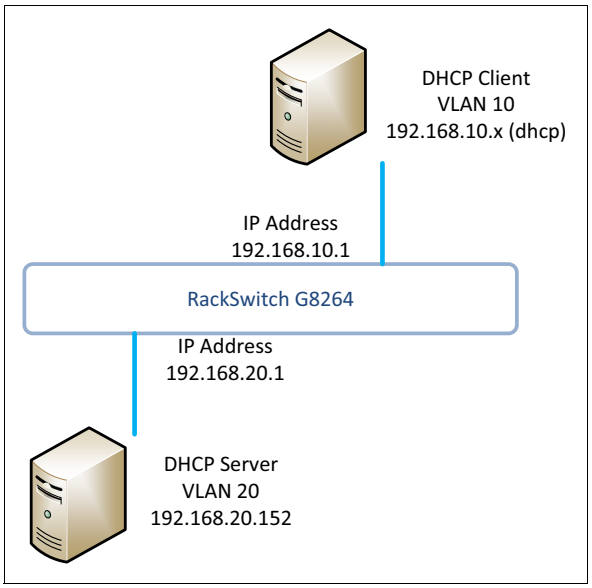


Figure 4-24 BootP and DHCP packet walk with a single switch

Example 4-5 shows how to correctly configure BootP and DHCP relay on a G8264 switch.

Example 4-5 DHCP Config example on a G8264

```
G8264-1#show run | section ip
interface ip 10
        ip address 192.168.10.1
        vlan 10
        enable
        exit
!
interface ip 20
        ip address 192.168.20.1
        vlan 20
        enable
        exit
!
ip bootp-relay server 1 address 192.168.20.152
ip bootp-relay enable
```

Example 4-6 shows DHCP relay counters to identify whether the request is being properly forwarded to the dhcp server and whether the response is being sent back to the client.

Example 4-6 DHCP counters for a successful request

```
G8264-1(config)#sh ip bootp counters
------------------------------------------------------------------
BOOTP Relay statistics for interface ip 10 :

Requests received from client:        2
Requests relayed to server:           2
Requests relayed with option 82:      0
Requests dropped due to ...
  - relay not allowed:                0
  - no server or unreachable server:  0
  - packet or processing errors:      0
Replies received from server:         0
Replies relayed to client:            2
Replies dropped due to ...
  - packet or processing errors:      0

BOOTP Relay statistics for interface ip 20 :

Requests received from client:        0
Requests relayed to server:           0
Requests relayed with option 82:      0
Requests dropped due to ...
  - relay not allowed:                0
  - no server or unreachable server:  0
  - packet or processing errors:      0
Replies received from server:         2
Replies relayed to client:            0
Replies dropped due to ...
  - packet or processing errors:      0
```

4.12.2 Layer 3 with VRRP and vLAG

VRRP enables redundant router configurations within a LAN, which provides alternative router paths for a host to eliminate single points-of-failure within a network. Each participating VRRP-capable routing device is configured with the same virtual router IPv4 address and ID number. One of the virtual routers is elected as the master (based on a number of priority criteria) and assumes control of the shared virtual router IPv4 address. If the master fails, one of the backup virtual routers takes control of the virtual router IPv4 address and actively processes traffic that is addressed to it.

If used in parallel with vLAG, VRRP performs Layer 3 routing on the master and backup switches. In this scenario, VRRP with BootP and DHCP relay uses the VRRP Address to communicate with the Client and Server.

Figure 4-25 shows a BootP and DHCP request process to a single switch.

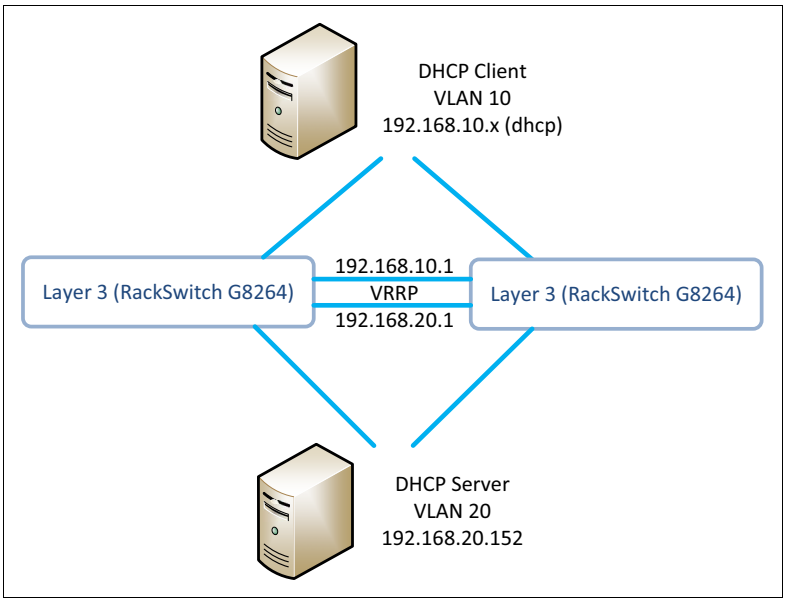


Figure 4-25 BootP and DHCP packet walk for a switch that is running VRRP with vLAG

Example 4-7 shows how to correctly configure BootP and DHCP relay with VRRP on a G8264 switch.

Example 4-7 DHCP Config example on a G8264 switch with VRRP configured

```
G8264-1(config)#show run | section ip
interface ip 10
        ip address 192.168.10.2
        vlan 10
        enable
        exit
!
interface ip 20
        ip address 192.168.20.2
        vlan 20
        enable
        exit
!
ip bootp-relay server 1 address 192.168.20.152
```

```
ip bootp-relay enable

G8264-1(config)#show run | section vrrp
router vrrp
        enable
!
        virtual-router 10 virtual-router-id 10
        virtual-router 10 interface 10
        virtual-router 10 priority 101
        virtual-router 10 address 192.168.10.1
        virtual-router 10 enable
!
        virtual-router 20 virtual-router-id 20
        virtual-router 20 interface 20
        virtual-router 20 priority 101
        virtual-router 20 address 192.168.20.1
        virtual-router 20 enable
```

Example 4-8 shows dhcp relay counters to identify whether the request is being properly forwarded to the dhcp server and whether the response is being sent back to the client. In this example, G8264-1 has an active connection to the dhcp client. G8264-2 has an active connection to the dhcp server to simulate a worst-case scenario by having to traverse the Interswitch link between the two G8264 Switches.

Example 4-8 DHCP counters of a successful request

```
G8264-1(config)#show ip bootp counters
------------------------------------------------------------------
BOOTP Relay statistics for interface ip 10 :

Requests received from client:        2
Requests relayed to server:           2
Requests relayed with option 82:      0
Requests dropped due to ...
  - relay not allowed:                0
  - no server or unreachable server:  0
  - packet or processing errors:      0
Replies received from server:         0
Replies relayed to client:            1
Replies dropped due to ...
  - packet or processing errors:      0

BOOTP Relay statistics for interface ip 20 :

Requests received from client:        0
Requests relayed to server:           0
Requests relayed with option 82:      0
Requests dropped due to ...
  - relay not allowed:                0
  - no server or unreachable server:  0
  - packet or processing errors:      0
Replies received from server:         1
Replies relayed to client:            0
Replies dropped due to ...
  - packet or processing errors:      0
```

4.13 Flex System Interconnect Fabric

Flex System Interconnect Fabric (Flex Fabric) is a multi-switch configuration that uses a pair of G8264CS switches as aggregators and a pair of SI4093 interconnect modules in each of up to nine Flex System chassis. Its underpinnings use the same features on the Broadcom switching ASICs as the Lenovo stacking implementation, and it has some of the same constraints. For more information about the features of the Flex Fabric, see *Flex System Interconnect Fabric*, TIPS1183, which is available at this website:

`http://lenovopress.com/tips1183`

The objective of the Flex Fabric is to enable construction of a multi-chassis pod that has a single attachment to a customer network, abundant east-west bandwidth, and appears as an end-node from the perspective of the upstream network. You can run one or more applications on the servers in the pod. In most instances, the servers run a hypervisor, such as VMware and applications that run on guest VMs can move to available servers as needed.

Key attributes of the Flex System Interconnect Fabric

The Flex Fabric pod often appears as shown in Figure 4-26 on page 94. It is configured as a unit much the same way as a stacked environment is configured, with the following exceptions:

- The G8264CS switches are the only devices that can be the master and the backup master and are identified as switch numbers 1 and 2.
- INT and EXT port aliases can be used for ports on the SI4093 modules, which are switch numbers that start at 3 and 4.
- The commands that include the option *stack* often are modified to use the word *fabric* instead.
- The **`bind fabric`** command automatically binds all SI4093 switches that are in *ATTACH* state and make them full members of the fabric, if possible; there is no need to identify them further.

Flex System Interconnect Fabric and alternatives

Flex Fabric competes with technologies including stacking and Easy Connect. These technologies can be used to make a pod of the type that is described in “Key attributes of the Flex System Interconnect Fabric” on page 92.

Flex System Interconnect Fabric and stacking

One key feature of the fabric implementation is the ability to perform a staggered reboot for a firmware upgrade or any other purpose. However, a stacked configuration is not subject to the following topology and other limitations that apply to Flex Fabric:

- There is only a single logical uplink to the core network.
- Servers cannot be attached to unused external ports on the G8264CS.
- Flex Fabric can use only G8264CS at top of rack and the SI4093 module in a chassis.
- All fabric links (switch-to-switch) must be the same bandwidth, which effectively limits them to 10 Gb ports.

An alternative pod configuration can use one or more stacks of EN4093 and CN4093 switches, and can aggregate the stacks by using G8264 or G8316 and G8332 switches at 40 Gb.

For more information about stacking, see 4.2, “Stacking” on page 60.

Flex System Interconnect Fabric and Easy Connect

Another option for creating a pod is to use Easy Connect with or without top-of-rack aggregation switches. Easy Connect has its own topology constraints, which are similar to those constraints of Flex Fabric. However, Easy Connect allows a choice of aggregation switches, which is not currently possible with Flex Fabric, and Easy Connect (or stand-alone SI4093 modules) can provide a pod that consists of a single chassis. The cost of the G8264CS switches might make Flex Fabric designs more attractive for environments that use FCoE with breakout at the ToR, but not so attractive to environments that are not using FCoE.

Configuration options and issues for Flex System Interconnect Fabrics

There are two options that can be used to connect an uplink group Flex System Interconnect Fabric (called *fabric* in the remainder of this section) to an upstream network. An uplink group is a set of VLANs and the links that are used to connect them to an upstream network; different uplink groups can choose different options. A VLAN can be part of a single uplink group only; this restriction is to ensure that there are no network loops in the upstream connection.

The following options are available:

- The use of a single PortChannel to carry one or more VLANs to the upstream network. The channel can be connected to multiple upstream switches if those switches support a distributed link aggregation function, such as vLAG, stacking, or Cisco vPC. It is not recommended to use an individual port when you are using this option because this port then constitutes a single point of failure.

  This option provides protection against the failure of all but one of the configured uplinks, and against failure of upstream switches if a feature, such as vLAG is used.

  To protect against a failure in one of those switches, ensure that the PortChannel includes ports from both of the G8264CS switches in the fabric.
- The use of a pair of ports or PortChannels with the *hotlinks* feature, which enables a backup port to be designated to carry traffic if a failure of the associated primary port or PortChannel occurs. If PortChannels are used with hotlinks, the primary and backup PortChannels can be connected to an upstream vLAG or similar environment.

  This option can be used in an environment where the upstream network does not provide a distributed link aggregation capability. It is analogous to active and standby NIC teaming because one port (or PortChannel) is active and the other acts as a backup. Ideally, the active and backup links are connected to different upstream switches. This configuration enables the environment to survive the failure of the upstream switch where the primary links are connected.

Diagrams of the two options and configuration excerpts to implement them are shown in Figure 4-26 on page 94 and Figure 4-28 on page 95.

Figure 4-26 on page 94 shows a Flex System Interconnect Fabric topology with LAG upstream connectivity.

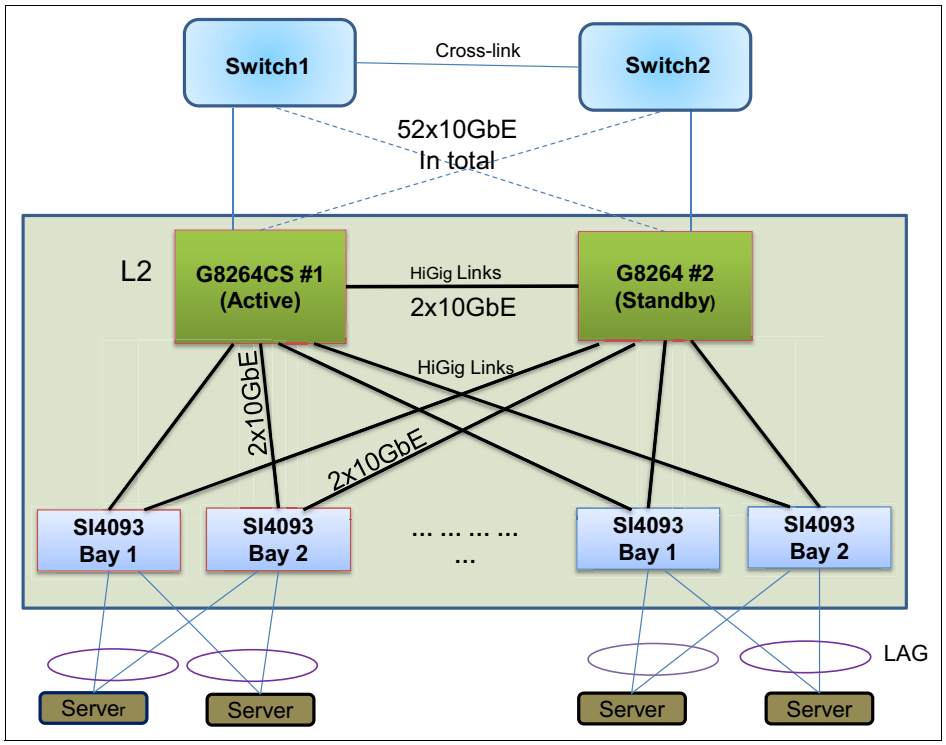


Figure 4-26 Flex System Interconnect Fabric topology with LAG upstream connectivity

Figure 4-27 shows an excerpt of the configuration for the topology that is shown in Figure 4-26.

```
interface port 1:40,1:41,2:40,2:41
        switchport mode trunk
        switchport trunk allowed vlan 10-11,201-202,2011,2022
        vlan dot1q tag native
        switchport trunk native vlan 10
        exit

interface port 1:40,1:41,2:40,2:41
        lacp mode active
        lacp key 40

Alternative to the LACP configuration that uses a static portchannel for same
ports:
portchannel 40 port 1:40,1:41,2:40,2:41 enable
```

Figure 4-27 Uplink configuration that uses a single LACP PortChannel

The configuration that is shown in Figure 4-27 on page 94 includes ports from both of the G8264CS switches as part of the uplink. It can survive a failure on either of those switches or their physical ports. The VLANs that are configured in this example can be changed to match the VLANs that are needed in a customer's environment.

Figure 4-28 shows uplink connectivity from the Flex System Interconnect Fabric to two upstream switches which do not share a single data plane and so only one is active.

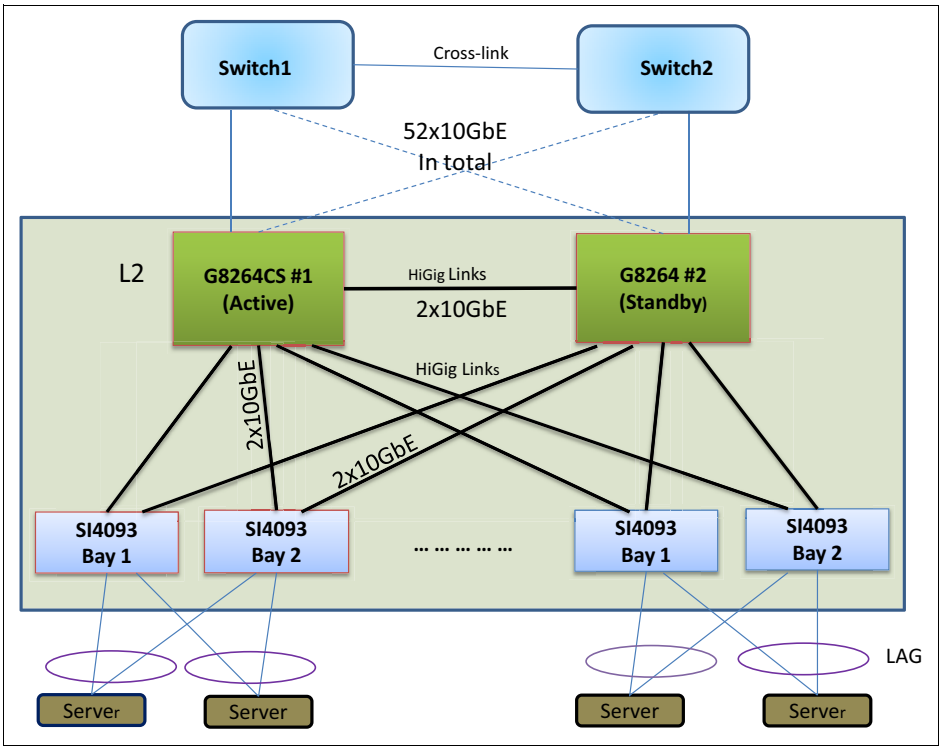


Figure 4-28 Flex System Interconnect Fabric with hotlinks active and standby uplinks

Figure 4-29 shows an excerpt of the configuration for the topology that is shown in Figure 4-28 on page 95.

```
interface port 1:45,1:46,2:45,2:46
        switchport mode trunk
        switchport trunk allowed vlan 4050,4070
        switchport trunk native vlan 4050
        exit

portchannel 100 port 1:45
portchannel 100 port 2:45
portchannel 100 enable
!
portchannel 200 port 1:46
portchannel 200 port 2:46
portchannel 200 enable

hotlinks fdb-update
hotlinks trigger 1 master PortChannel 100
hotlinks trigger 1 backup PortChannel 200
hotlinks trigger 1 forward-delay 5
no hotlinks trigger 1 backup prefer
hotlinks trigger 1 enable
!
hotlinks enable
```

Figure 4-29 Uplink configuration with hotlinks and active and standby PortChannels

The configuration of the hotlinks feature that is shown in Figure 4-29 on page 96 passes through many validity checks that make it difficult to configure without triggering an error message. A technique that is called *delayed apply* mode allows the entire configuration section to be entered and then checked for consistency only after a logical point is reached. Use the iscli mode **`delayed-apply`** command to enter this mode. Configuration commands can then be entered and are validated when the **`apply`** command is entered. At the time of this writing, delayed apply is available with the Flex System Interconnect Fabric only.

As shown in Figure 4-29, the hotlinks feature uses the ports on PortChannel 100 to carry the configured VLANs (4050 and 4070) if at least one of them is up. If PortChannel 100 fails completely, PortChannel 200 (shown with dotted lines in Figure 4-28 on page 95) carries traffic for all VLANs. A sequence of bridge update packets is sent to the upstream switches so that their forwarding tables are quickly updated to reflect this.

The **`forward-delay`** option that is configured in Figure 4-29 on page 96 is the number of seconds to wait before transitioning from the master to the backup links. This feature is provided to protect against link instability (often called *link flapping*), and has a default setting of 1 second. During this interval (which begins when the primary link failure is detected and ends after the specified number of seconds), traffic might not be forwarded or received on the master or backup links.

VLAN load balancing with hotlinks

There is a feature that is added to hotlinks that uses the master and backup links and splits the VLANs that are being carried by the links so that approximately half of the VLANs are carried on the configured master links and the remainder are carried on the backup links. If either uplink experiences an outage, the other links carry traffic for all VLANs. Which VLANs are carried by which set of links also can be explicitly selected.

To simplify the environment, this feature is disabled in the configuration fragment that is shown in Figure 4-29 on page 96. Use the **`no hotlinks trigger 1 backup prefer`** CLI command to disable this feature. The command must be entered separately if multiple trigger groups are used. If you use this feature, it is enabled in the default settings. You might need to modify to the configurations of the upstream switches if you use this feature.

5

Layer 3 technologies

This chapter includes preferred practices for implementing common Layer 3 technologies with some of the options in the Lenovo Networking products. It is not intended to describe all capabilities, but it refers to more information about these technologies.

This chapter includes the following topics:

5.1 OSPF with VRRP and vLAG

Open Shortest Path First (OSPF) is one of the most common enterprise Layer 3 technologies that are used within a data center for dynamic distribution of Layer 3 routes in a medium to large environment. OSPF has many advantages that use several operational characteristics to makes it efficient.

For more information about OSPF and its capabilities within the Lenovo Networking Products, see the Application and isCLI guides that are available at this website:

http://ibm.com/support/entry/portal/Documentation

When OSPF and Virtual Router Redundancy Protocol (VRRP) are introduced on a pair of Lenovo Layer 3-capable switches, a highly redundant environment is created that allows for a floating Layer 3 IP Gateway Address. VRRP allows for various options that include the ability to define how and when a Layer 3 IP elect to change ownership between the pair of redundant switches. VRRP was originally developed to act as a Master/Slave relation where the Master switch is the only switch that responds to client ARP requests.

When vLAG is introduced with VRRP, this configuration enables active/active teaming of a third device and VRRP to be active/active.

Figure 5-1 shows an environment with all three options that are configured in which both switches allow for OSPF and VRRP and can act as Layer 3 active devices with the enabled vLAG ports. By allowing for both switches to act as a Layer 3 device, traffic can be split across both active pairs to reduce the total amount of bandwidth on any one Layer 3 switch.

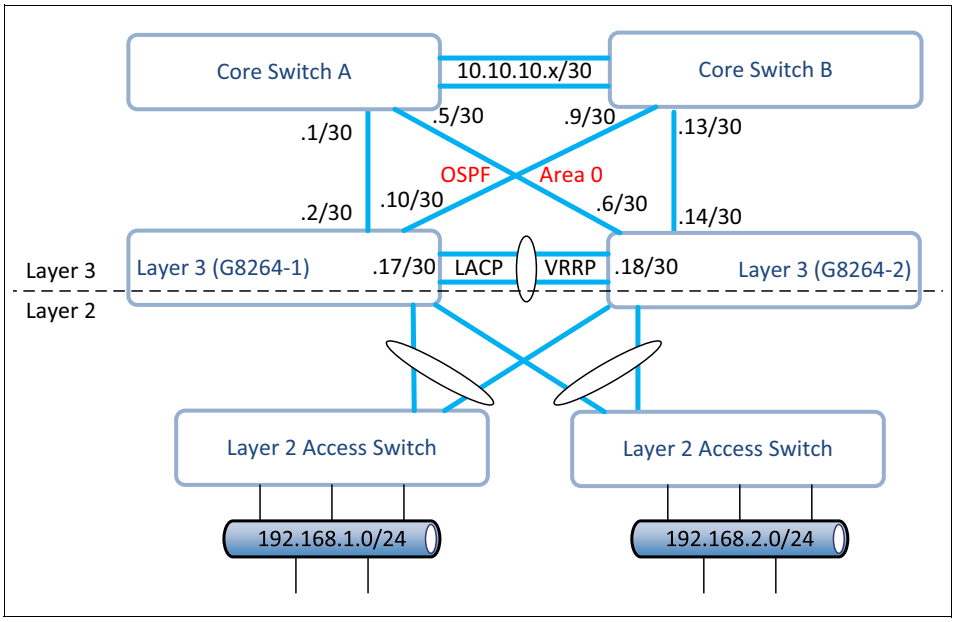


Figure 5-1 OSPF with VRRP and vLAG

Example 5-1 shows an entire **show run** of the G8264-1 switch that is shown in Figure 5-1 on page 100.

Example 5-1 Output of a **show run**

```
G8264-1(config)#show run
Current configuration:
!
version "7.9.10"
switch-type "Networking Operating System RackSwitch G8264"
iscli-new
!
no system default-ip
hostname "G8264-1"
!
interface port 1
        description "ISL/Peer-Link"
        switchport mode trunk
        switchport trunk allowed vlan 10,20,30,4090
        switchport trunk native vlan 4090
        spanning-tree guard loop
        exit
!
interface port 5
        description "ISL/Peer-Link"
        switchport mode trunk
        switchport trunk allowed vlan 10,20,30,4090
        switchport trunk native vlan 4090
        spanning-tree guard loop
        exit
!
interface port 17
        description "ESXi111"
        switchport mode trunk
        switchport trunk allowed vlan 10,20,30
        switchport trunk native vlan 10
        bpdu-guard
        spanning-tree portfast
        exit
!
interface port 18
        description "ESXi112"
        switchport mode trunk
        switchport trunk allowed vlan 10,20,30
        switchport trunk native vlan 10
        bpdu-guard
        spanning-tree portfast
        exit
!
interface port 19
        description "dhcp-client"
        shutdown
        switchport access vlan 10
        bpdu-guard
        spanning-tree portfast
        exit
!
interface port 20
        description "dhcp-server"
        switchport access vlan 20
        bpdu-guard
```

```
        spanning-tree portfast
        exit
!
interface port 29
        description "G8000"
        switchport mode trunk
        switchport trunk allowed vlan 10,20,30,999
        switchport trunk native vlan 999
        exit
!
interface port 40
        switchport mode trunk
        switchport trunk allowed vlan 1,200
        exit
!
interface port 63
        description "Core-c3560-Gi0/1"
        no switchport
        ip address 10.10.10.2 255.255.255.252 enable
        ip ospf enable
        exit
!
interface port 64
        description "Core-c3560-Gi0/3"
        no switchport
        ip address 10.10.10.10 255.255.255.252 enable
        ip ospf enable
        exit
!
vlan 10
        name "MGMT-NET"
!
vlan 20
        name "vMotion"
!
vlan 30
        name "DATA-NET"
!
vlan 999
        name "Native"
!
vlan 4090
        name "ISL-Trunk"
!
spanning-tree stp 1 vlan 1
!
spanning-tree stp 10 bridge priority 4096
spanning-tree stp 10 vlan 10
!
spanning-tree stp 20 bridge priority 4096
spanning-tree stp 20 vlan 20
!
no spanning-tree stp 26 enable
spanning-tree stp 26 vlan 4090
!
spanning-tree stp 30 bridge priority 4096
spanning-tree stp 30 vlan 30
!
spanning-tree stp 110 bridge priority 4096
spanning-tree stp 110 vlan 999
```

```
!
interface port 1
        lacp mode active
        lacp key 105
!
interface port 5
        lacp mode active
        lacp key 105
!
interface port 29
        lacp mode active
        lacp key 1029
!
vlag enable
vlag tier-id 10
vlag hlthchk peer-ip 192.168.0.92
vlag isl adminkey 105
vlag adminkey 1029 enable
!
ip router-id 1.1.1.1
!
interface ip 10
        ip address 192.168.10.2
        vlan 10
        enable
        exit
!
interface ip 20
        ip address 192.168.20.2
        vlan 20
        enable
        exit
!
interface ip 30
        ip address 192.168.30.2
        vlan 30
        enable
        exit
!
interface ip 100
        ip address 10.10.10.17 255.255.255.252
        vlan 4090
        enable
        exit
!
!interface ip 128
!       addr <dhcp>
!
!ip gateway 4 addr <dhcp>
!ip gateway 4 enable
!
router vrrp
        enable
!
        virtual-router 10 virtual-router-id 10
        virtual-router 10 interface 10
        virtual-router 10 priority 101
        virtual-router 10 address 192.168.10.1
        virtual-router 10 enable
!
```

```
        virtual-router 20 virtual-router-id 20
        virtual-router 20 interface 20
        virtual-router 20 priority 101
        virtual-router 20 address 192.168.20.1
        virtual-router 20 enable
!
        virtual-router 30 virtual-router-id 30
        virtual-router 30 interface 30
        virtual-router 30 priority 101
        virtual-router 30 address 192.168.30.1
        virtual-router 30 enable
!
router ospf
        enable
        area 0 enable
        redistribute fixed export 1 1
!
interface ip 100
        ip ospf enable
        ip ospf priority 255
!
end
```

5.2 BGP with VRRP and vLAG

Although Border Gateway Protocol (BGP) is most common in the wide area network, it also has its merits in an enterprise Layer 3 technology that is used within some of the larger data centers that are often used to divide between boundaries.

For more information about BGP and its capabilities within Lenovo Networking products, see the Application and isCLI guides that are available at this website:

http://ibm.com/support/entry/portal/Documentation

When BGP and VRRP are introduced on a pair of Layer 3-capable switches, a highly redundant environment is created for a floating Layer 3 IP Gateway Address. VRRP enables various options to define how and when a Layer 3 IP elects to change ownership between a pair of redundant switches. VRRP was originally developed to act as a Master/Slave relation where the Master switch is the only device that responds to client ARP requests.

When introducing vLAG with VRRP, this configuration enables active/active teaming of a third device and VRRP to act as an active/active Layer 3 pair of switches.

Figure 5-2 shows an environment with all three options that are configured in which both switches allow for BGP and VRRP and the switches can act as Layer 3 active devices with the enabled vLAG ports. By allowing for both switches to act as a Layer 3 device, traffic can be split across both active pairs to reduce the total amount of bandwidth on any one Layer 3 switch.

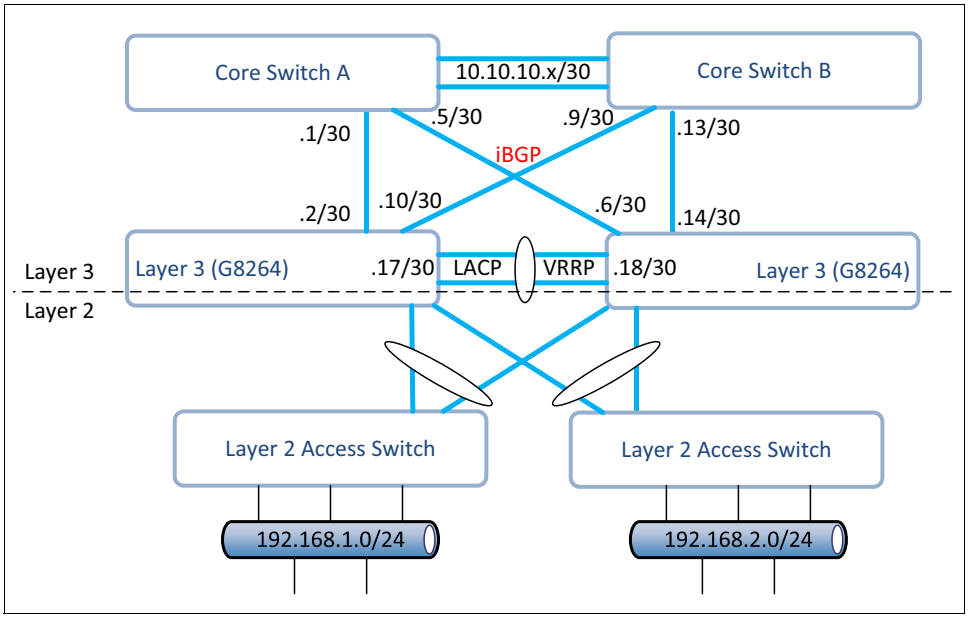


Figure 5-2 BGP and VRRP with vLAG

The configuration of a BGP implementation is similar to that of an OSPF implementation except for defining an eBGP or iBGP relationship with its peers.

Example 5-2 shows the BGP portion of the configuration only. The rest of the configuration is described 5.1, “OSPF with VRRP and vLAG” on page 100.

Example 5-2 Output of the iBGP portion of a show run

```
router bgp
        as 62001
        enable
!
router bgp
        no neighbor 1 shutdown
        neighbor 1 remote-address 10.10.10.1
        neighbor 1 remote-as 61001
        neighbor 1 redistribute fixed
!
router bgp
        no neighbor 2 shutdown
        neighbor 2 remote-address 10.10.10.9
        neighbor 2 remote-as 61009
        neighbor 2 redistribute fixed
!
ip router-id 1.1.1.1
```

5.3 ECMP with static and dynamic routes

Equal Cost Multipath (ECMP) provides load sharing across multiple static or dynamic routes to a single destination. Figure 5-3 shows how dynamic ECMP routes can be used to loadshare traffic.

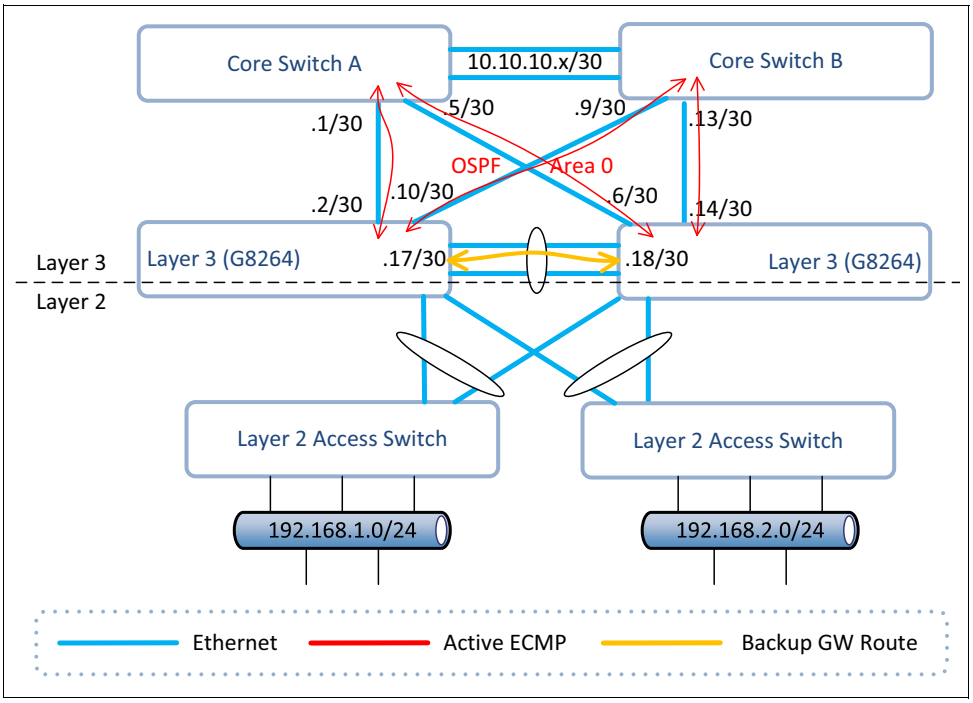


Figure 5-3 ECMP that uses dynamic routes

Static routes specify to which IP source network, mask, and destination address to send traffic. A second route with the same source network and mask but with a different destination address allows a Layer 3 switch to split the traffic across the multiple routes based on source IP, destination IP, Layer 4 port, and protocol number (by default). For more information about static routes, see 5.5.2, “Static Routing with vLAG” on page 111.

Example 5-3 shows two static routes that use the same source network and mask to two separate gateway addresses, which enables load share traffic.

Example 5-3 ECMP that uses static route configuration

```
!
ip route 10.10.10.0 255.255.255.0 192.168.50.1
ip route 10.10.10.0 255.255.255.0 192.168.60.1
ip route healthcheck
!
```

Dynamic routes can use ECMP by specifying multiple routes to the same physical destination and loadshare based on IP Source, Destination, Layer 4 port, and protocol number (by default). For more information about dynamic routes, see 5.5.1, “Dynamic Routing with vLAG” on page 110.

Example 5-4 on page 107 shows the configuration of two dynamic routes that use the same source network and mask to two separate gateway addresses, which enables loadshare traffic.

Example 5-4 ECMP that uses dynamic route configuration

```
interface port 63
        description "Core-c3560-Gi0/1"
        no switchport
        ip address 8.8.8.2 255.255.255.252 enable
        ip ospf enable
        exit
!
interface port 64
        description "Core-c3560-Gi0/3"
        no switchport
        ip address 10.10.10.2 255.255.255.252 enable
        ip ospf enable
        exit
!
interface ip 7
        ip address 7.7.7.1 255.255.255.252
        vlan 4090
        enable
        exit
!
ip router-id 1.1.1.1
!
router ospf
        enable
        area 0 enable
        redistribute fixed export 1 1
!
interface ip 7
        ip ospf enable
        ip ospf priority 255
```

Example 5-5 shows the output of a **show ip ospf neighbor** command that includes a VLAN-based neighbor that runs across the peer-link/ISL PortChannel (by using VLAN 4090 [native VLAN] with STP disabled) and two Routed Port OSPF neighbors that are directly connected.

Example 5-5 Output of a show ip ospf neighbor and a show ip route tag ospf

```
G8264-1(config)#show ip ospf neighbor
Intf  NeighborID       Prio  State         Address
----  ----------       ----  -----         -------
   7  1.1.1.2           255  Full           7.7.7.2

Routed Port OSPF Neighbors:
Port  NeighborID       Prio  State         Address
----  ----------       ----  -----         -------
  63  192.168.0.104       1  Full           8.8.8.1
  64  192.168.0.104       1  Full           10.10.10.1

G8264-1(config)#show ip route tag ospf
Mgmt routes:
Status code: * - best
    Destination         Mask          Gateway        Type      Tag      Metric   If
  --------------- --------------- --------------- --------- --------- ---------- --

Data routes:
```

```
Status code: * - best
    Destination         Mask          Gateway        Type      Tag      Metric   If
  --------------- --------------- --------------- --------- --------- ---------- --
* 0.0.0.0         0.0.0.0         10.10.10.1      indirect  ospf              1 routed
* 0.0.0.0         0.0.0.0         8.8.8.1         indirect  ospf              1 routed
* 9.9.9.0         255.255.255.252 7.7.7.2         indirect  ospf              2  7
* 9.9.9.0         255.255.255.252 10.10.10.1      indirect  ospf              2 routed
* 9.9.9.0         255.255.255.252 8.8.8.1         indirect  ospf              2 routed
  192.168.10.0    255.255.255.0   7.7.7.2         indirect  ospf              2  7
  192.168.20.0    255.255.255.0   7.7.7.2         indirect  ospf              2  7
  192.168.30.0    255.255.255.0   7.7.7.2         indirect  ospf              2  7
```

The neighbor address of 7.7.7.2 is not within the available list of default destination routes. This address is unavailable because the configuration for the OSPF interface (which is used as the peer-link/ISL) has a priority cost of 250. This priority causes the OSPF default route to not be used unless all directly connected equal cost uplinks to the core fail. Only then is the peer-link/ISL injected into the routing table as a default route.

Example 5-6 shows a failure of both uplinks to the core that leaves only the peer-link/ISL available to be used.

Example 5-6 Output of a show ip route tag ospf

```
G8264-1(config)#show ip route tag ospf
Mgmt routes:
Status code: * - best
    Destination         Mask          Gateway        Type      Tag      Metric   If
  --------------- --------------- --------------- --------- --------- ---------- --

Data routes:
Status code: * - best
    Destination         Mask          Gateway        Type      Tag      Metric   If
  --------------- --------------- --------------- --------- --------- ---------- --
* 0.0.0.0         0.0.0.0         7.7.7.2         indirect  ospf              1  7
* 9.9.9.0         255.255.255.252 7.7.7.2         indirect  ospf              2  7
  192.168.10.0    255.255.255.0   7.7.7.2         indirect  ospf              2  7
  192.168.20.0    255.255.255.0   7.7.7.2         indirect  ospf              2  7
  192.168.30.0    255.255.255.0   7.7.7.2         indirect  ospf              2  7
```

5.4 Route maps

Route maps are used to filter routes to neighboring switches. Although it is possible to enable a redistribution process, route maps can manipulate which routes are advertised or filtered out to and from a remote pier. Route maps can be used for OSPF and BPG processes.

Example 5-7 shows an eBGP route that is injected into an OSPF process by using a route map. When a route map is defined for a particular redistribution process, only those routes within the IP match-address list are advertised and all other routers are ignored.

Note: The redistribution command also must be enabled if you intend to filter advertisements.

Example 5-7 Route map to advertise (eBGP route into OSPF) 10.10.10.1 255.255.255.224 only

```
!
ip match-address 1 10.10.10.1 255.255.255.224
ip match-address 1 enable
!
route-map 1
        no local-preference
        precedence 10
        metric 10
        metric-type 1
        no weight
        access-list 1 match-address 1
        no access-list 1 metric
        access-list 1 action permit
        access-list 1 enable
!
router ospf
        area 0 enable
        area 0 area-id 0.0.0.0
        area 0 type transit
        area 0 stub-metric 1
        no area 0 authentication-type
        area 0 spf-interval 10
!
        redistribute ebgp 1
        redistribute fixed 1 1
!
```

Example 5-8 on page 110 shows a BGP route map of a fixed interface. Although there might be 10 - 15 fixed interfaces that are defined to support the local Layer 2 environment, you might want to advertise only a few of them to the remote BGP pier or split your advertisements among several BGP piers.

Example 5-8 Route map to advertise (BGP route to neighbor) 10.10.10.1 255.255.255.224

```
!
ip match-address 1 10.10.10.1 255.255.255.224
ip match-address 1 enable
!
!
route-map 1
        no local-preference
        precedence 10
        no metric
        no metric-type
        no weight
        access-list 1 match-address 1
        no access-list 1 metric
        access-list 1 action permit
        access-list 1 enable
        enable
!
router bgp
        no neighbor 1 shutdown
        neighbor 1 remote-address 9.9.9.2
        neighbor 1 remote-as 55556
        neighbor 1 timers hold-time 90
        neighbor 1 timers keep-alive 30
        neighbor 1 route-map out 1
        neighbor 1 redistribute fixed
!
```

5.5 Layer 3 with vLAG and limitations

Static and Dynamic Layer 3 with VRRP and vLAG are all supported on the same pair of switches at the same time. Layer 3 VRRP with vLAG can provide for an active/active environment on the Layer 2 and Layer 3 portions of the Switch pair. Although it is still preferred practice to enable and use Spanning Tree to prevent Layer 2 loops, having vLAG enabled can reduce or eliminate the number of blocked paths, which effectively reduces the requirements for Spanning Tree. However, it is still advisable to have Spanning Tree enabled to help prevent any future potential broadcast storms.

5.5.1 Dynamic Routing with vLAG

Dynamic Routing, such as OSPF and BGP, are supported by vLAG on the same pair of switches. However, vLAG cannot be a member of a port that also is peering with a neighbor. For example, if OSPF (or BGP) is used on a port to form an adjacency with a neighbor, it is not supported to also enable vLAG on that specific port.

Figure 5-4 shows the use of dynamic routes and vLAG on the same pair of switches. The OSPF point-to-point connections are single port adjacencies while the Layer 2 portion in this example is using PortChannels and vLAG to provide an even distribution to both vLAG enabled Layer 3 G8264 switches.

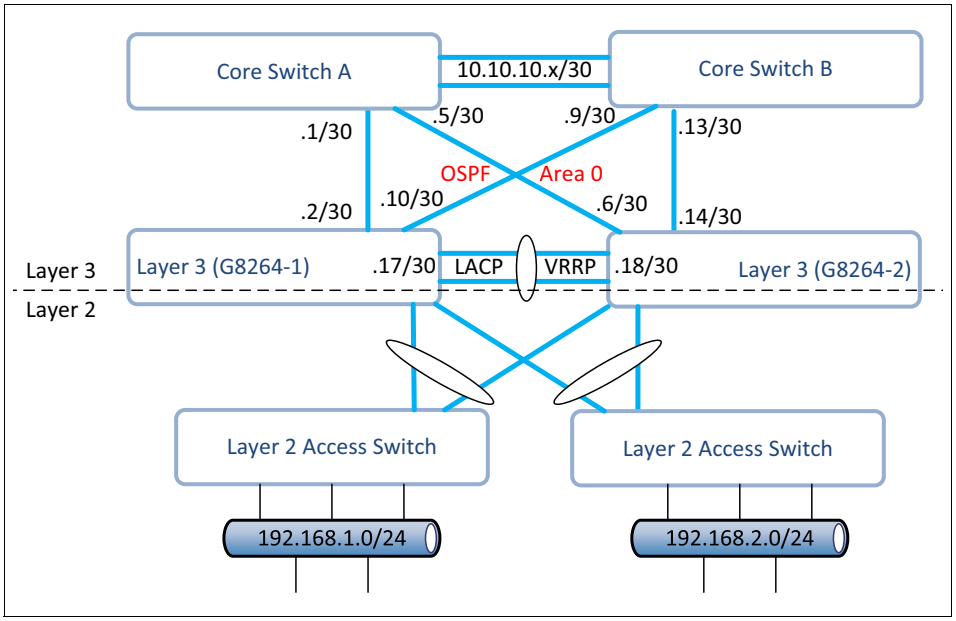


Figure 5-4 Pair of G8264s with vLAG and Dynamic Layer 3 routing enabled

5.5.2 Static Routing with vLAG

Unlike Dynamic routing, Static routing can support vLAG on the same switch ports. This support means that a pair of vLAG enabled switches with VRRP and default gateways that point upstream towards the core can also be connected to the core via a pair of vLAG ports to provide for Layer 2 and Layer 3 active/active across the same set of ports.

Figure 5-5 shows the use of static routes and vLAG over the same ports.

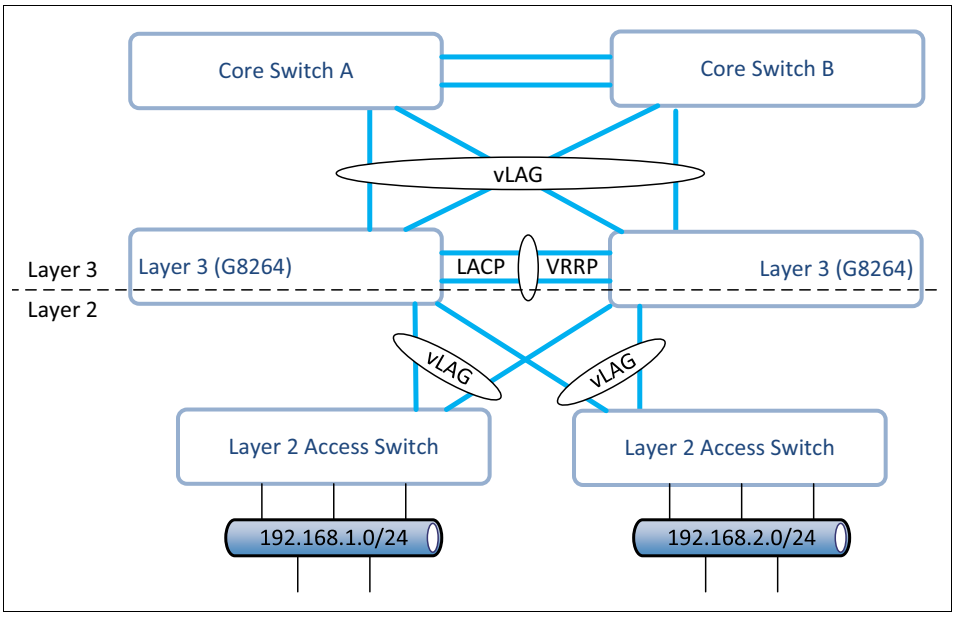


Figure 5-5 Pair of G8264s with vLAG and Static Layer 3 routing enabled

5.5.3 Spanning Tree with vLAG

Spanning Tree with vLAG is supported on the same pair of switches if it is a form of multi-Spanning Tree. For example, although MSTP and PVRST+ support vLAG, RSTP and vLAG cannot be enabled on the same switch at the same time.

The following parameters also are available in the 7.8 version of the *G8264 Application Guide*:

- When VLAG is configured or changes are made to your VLAG configuration, consider the following VLAG behavior:
  - When a static Mrouter is added on VLAG links, ensure that you also add the Mrouter on the ISL link to avoid VLAG link failure. If the VLAG link fails, traffic cannot be recovered through the ISL.
  - When you enable VLAG on the switch, the ISL shuts down if an MSTP region mismatch is detected with the VLAG peer. In such a scenario, correct the region on the VLAG peer and manually enable the ISL.
  - If you enabled VLAG on the switch and you must change the STP mode, ensure that you first disable VLAG and then change the STP mode.
  - When VLAG is enabled, you might see two root ports on the secondary VLAG switch. One of these ports is the actual root port for the secondary VLAG switch and the other port is a root port that is synced with the primary VLAG switch.
  - The LACP key used must be unique for each VLAG in the entire topology.
  - The STG to VLAN mapping on both VLAG peers must be identical.

- The following parameters must be identically configured on the VLAG ports of both of the VLAG peers:
  - VLANs
  - Native VLAN tagging
  - STP mode
  - BPDU Guard setting
  - STP port setting
  - MAC aging timers
  - Static MAC entries
  - ACL configuration parameters
  - QoS configuration parameter

6

Securing access to the switch

There are numerous strategies that can be used to secure your switch. In this chapter, we describe these strategies in the preferred practice order of priority. The security features range from user authentication to protocols that are used and which remote systems can manage the switch.

This chapter includes the following topics:

6.1 Local User Authentication

All of the top-of-rack (ToR) and embedded switches support the configuration local users. Even if local user authentication is not the primary source of access control, it is important to understand and configure local users because they are used when remote authentication servers are down.

User access is allocated by using the following levels of access permissions with decreasing levels of authority:

- Administrator

  The Administrator (super user) level of authority permits switch configuration and all of the privileges of the lower users (oper and user).

- Operator

  An Operations user permits operations activities (operationally enable or disable ports, reboot, and so on) and commands that are available to user.

- User

  A user level is the lowest authority level and allows the user to display counters and state information.

The local user authentication commands are all contained under the access user command set. There are built in user names with the names identified as user privilege levels (`admin`, `oper`, or `user`) and locally defined user name with assigned privilege levels. By default, the user name `admin` with the password `admin` is enabled on all switches with another user named USERID with the password PASSW0RD (zero for the O) on the Flex System embedded switches. The user name `user` with password `user` is also enabled in early Networking Operating System (NOS) releases. Use the **`show access user`** command to determine the currently configured and enabled users. Example 6-1 shows an example of the command output.

Example 6-1 Output for the show access user command

```
Usernames:
  user     - disabled   - offline
  oper     - disabled   - offline
  admin    - enabled    - online     1 session.
Current User ID table:
  1: name USERID  , ena, cos admin   , password valid, offline

Current strong password settings:
  strong password status: disabled
```

The following information is required to create a locally defined user:

- Valid user name
- Valid password
- Class of user (*admin/oper/user*)

Example 6-2 shows the commands to create a locally defined administrator user.

Example 6-2 Commands to create a locally defined administrator user

```
enable
configure terminal
access user 2 name NEWUSER
access user 2 password
```

Note: You are prompted for the admin password and the new user password.

```
access user 2 level administrator
access user 2 enable
```

The configuration that is shown in Example 6-2 yields the configuration that is shown in Example 6-3 (the output from **show running-config | include "access user 2"**).

Example 6-3 Output from the show running-config | include "access user 2" command

```
access user 2 name "NEWUSER"
access user 2 password
"cb1a087f480a082a9672e3b68bbaabd8ec3f61483ba040fb33058a55f0bbe204"
access user 2 enable
access user 2 level admin
```

The password in the configuration is encrypted and can be pasted into another switch to yield the configured password.

When you are considering locally administered users, consider the following preferred practices:

- Change the `admin` user password by logging in as `admin` and by using the **password** command or the **administrator-password** command.
- If the default user USERID is enabled, change that password by using the **access user 1 password** command and consider changing the user name by using the **access user 1 name** command.
- If the built-in user name `user` is enabled, disable it by entering a new user password as an empty string by using the **access user user-password** command.
- Add an administrator user name as a backup administrator account if you enable user lockout.
- Enable strong passwords to enforce improved security with expiry and lockout, as shown in the following example:

  ```
  access user strong-password enable
  access user strong-password lockout
  ```

  Optionally, you can configure the following settings:

  ```
  access user strong-password expiry  60(default = 120 days)
  access user strong-password faillock 5(default = 3 failures)
  ```

6.2 TACACS authentication

Terminal Access Controller Access Control System (TACACS) is the most commonly used remote authentication system for networking devices. This method enables user authentication, command authorization, and accounting services, which is why Cisco categorizes this system as AAA. The most commonly used feature of TACACS is the authentication feature, which provides centralized management of users. By using this feature, you use the same user name and password on every switch.

For TACACS configuration, the following information is required:

- TACACS Server address. The server can be configured by the IP address or the DNS name (if DNS is configured on the switch).
- Shared security key. This key is a character string that is configured on the TACACS server and must match with the configured value. This key is used for encrypting the data that is exchanged between the switch and the TACACS server. When the running configuration is displayed, the key that is entered is converted to an encrypted key that can be copied into other switches.
- Understanding of the privilege level mapping. The default privilege levels on the switches are not the same as those for Cisco switches. You can match the Cisco privilege levels by using the **`tacacs-server privilege-mapping`** command. The switch's current privilege levels can be displayed by using the **`show tacacs-server`** command. Example 6-4 shows the default mapping for the switch.

Example 6-4 Default mapping for the switch

```
Remote privilege  Local mapping
 ----------------  -------------
  0                  user
  1                  not set
  2                  not set
  3                  oper
  4                  not set
  5                  not set
  6                  admin
  7                  not set
  8                  not set
  9                  not set
 10                  not set
 11                  not set
 12                  not set
 13                  not set
 14                  not set
 15                  not set
```

As you can see, the `admin` and `oper` privilege levels are mapped to 6 and 3 in the default settings. The **`tacacs-server privilege-mapping`** command yields the mapping that is shown in Example 6-5 on page 119. It is compatible with Cisco (which is used most often).

Example 6-5 Output of the tacacs-server privilege-mapping command

```
Remote privilege  Local mapping
 ----------------  -------------
  0                 user
  1                 user
  2                 not set
  3                 not set
  4                 not set
  5                 not set
  6                 oper
  7                 oper
  8                 oper
  9                 not set
 10                 not set
 11                 not set
 12                 not set
 13                 not set
 14                 admin
 15                 admin
```

Example 6-6 shows an example TACACS configuration.

Example 6-6 Preferred practice TACACS configuration

```
enable
configure terminal
tacacs-server primary-host 172.16.6.101 key MySecret
tacacs-server secure-backdoor
tacacs-server privilege-mapping
tacacs-server enable
```

The **tacacs-server secure-backdoor** command that is shown in Example 6-6 is a preferred setting to enable a back door method to log in to the switch only when the TACACS server is not responding. In the default settings, the back door is always enabled on the serial console port, which can be considered a security threat. To use the TACACS back door, enter the user name `notacacs` when prompted for the user name. You are then prompted for a locally administered user name and password. When the `notacacs` user name is entered in earlier firmware releases, you are prompted to enter the TACACS password. Enter the locally configured `admin` for the back door login.

Keep a login session open until the TACACS authentication is validated so that any corrections to the configuration is made. Also, temporarily override the idle timeout of 5 minutes by using the **system idle 0** command so that this session does not time out. If you do not change the timeout value, you can be locked out of the switch and reset the switch to factory defaults to recover.

6.3 Management protocols

Several management protocols are available for switch administration. Table 6-1 shows what is enabled in the default settings and what to enable as a preferred practice.

Table 6-1 Management protocols

Protocol	TOR and BladeCenter	Flex System	Preferred practice
Telnet	Enabled	Disabled	Disabled
Secure Shell (SSH)	Disabled	Enabled	Enabled
HTTP	Enabled	Disabled	Disabled
HTTPS	Disabled	Enabled	Enabled
SNMP	Read-Write	Read-Write	Read-Only

These protocols can be controlled by using access commands and the current configuration can be displayed by using the command show access. It is a preferred practice to use secure protocols command line and web access. Many security departments consider SNMP write as a security issue; therefore, it is preferred to configure it as read-only.

Example 6-7 shows the preferred commands to configure access.

Example 6-7 Commands to configure access

```
enable
configure terminal
access snmp read-only
access https enable
no access http enable
ssh enable
no access telnet enable
```

The command to disable telnet is last because any open telnet session is closed when the command is entered.

6.4 SSH public key

SSH public keys are used to improve security when SSH are used by requiring a more complex authentication method than a simple password. SSH public keys are 768 - 2048 bits with the large keys being more secure. A 1024-bit key often is sufficient to secure most links. Another advantage of SSH public keys is that it can allow you to log in to the switch by using SSH without being prompted for user name and password, which can be useful in scripting.

The SSH keys have a public key that is shared by the client and server with the switch being the server and a private key that is used by the client. For SSH keys to work, you must first create a user name to be associated with the key and then copy the pubic key into the switch and associate it with a user name. The keys are created on the client systems. For example, with PuTTY, you can use the PuTTY Key Generator (PuTTYgen), which is a part of the PuTTY distribution.

Example 6-8 shows the steps that can be used for loading a key that is named `MyPublicKey` for the user name `NEWUSER`.

Example 6-8 Example steps for loading a key

```
enable
configure terminal
access user 2 name NEWUSER
access user 2 password
NOTE: You will be prompted for the admin password and the new user password.
access user 2 level administrator
access user 2 enable
copy tftp public-key address 172.16.6.101 filename MyPublicKey username NEWUSER
```

Use the **`show ssh-clientpubkey all`** command to verify the keys.

6.5 Restricting the devices with management access by using MNet/MMask

The Management Network commands are useful to increase the security of the switches by defining ranges of IP addresses that can access the switches to run management features. The **`access management-network`** commands is used to configure IPv4 access and the **`access management-network6`** command is used to configure IPv6 with each controlling access to the telnet, SSH, and BBI (Web) access or SNMP. Each enable the configuration of a range of addresses by using a mask for IPv4 or a prefix length for IPv6.

This feature can be useful if you want to restrict the management from one or more “jump boxes” or restricting the management to ranges of allowed address. When this feature is used, take care to create a management network entry that allows the device that you are using to have management access so that you avoid inadvertently locking yourself out of the switch.

The following commands enable access from a single IPv4 address for full SNMP access and access to the other management features:

```
enable
configure terminal
access management-network 172.16.6.101 255.255.255.255
access management-network 172.16.6.101 255.255.255.255 snmp-rw
```

The following commands enable access to a full subnet:

```
enable
configure terminal
access management-network 172.16.5.0 255.255.255.0
access management-network 172.16.5.0 255.255.255.0 snmp-rw
```

6.6 Management Access Control Lists

Management Access Control Lists (MACLs) are used to filter traffic that is destined to the switch's CPU from the data path where generic Access Control Lists act on traffic that is passing through the data path. MACLs are useful to allow or deny any traffic to the CPU instead of to only the management traffic with Management Networks so they can protect the CPU against many attacks. MACLs are supported on every switch except the G8124 with up to 256 available. The MACLs are processed from the lowest ACL number to higher so when c MACLs are created, you want to add a default deny action as the highest MACL.

To configure a MACL, you must define the ACL characteristics and enable it. Example 6-9 shows a MACL to allow everything from the subnet 172.16.0.0 and to deny everything else.

Example 6-9 MACL to allow everything from the subnet 172.16.0.0 and to deny everything else

```
enable
configure terminal
access-control macl 1 ipv4 source-ip-address 172.16.0.0 255.255.0.0
access-control macl 1 action permit
access-control macl 1 enable
access-control macl 256 ipv4 source-ip-address 0.0.0.0 0.0.0.0
access-control macl 256 action deny
access-control macl 256 enable
```

6.7 Password recovery

If the administrator password is lost, the only accepted recovery method is to reset the switch to the factory default. This reset can be accomplished through the boot menu via the serial console by pressing **Shift+B** while the memory test is running during the switch boot. Alternatively, the switch can be reset to default by pressing and holding the reset button on the switch's faceplate until all of the LEDs flash.

After the switch is reset, you can log in by using the default user name (`admin`) and password (`admin`). The switch configuration can be recovered from the active configuration block by using the **`show active-config`** command. The configuration can be copied into a text editor and the line that contains the **`access user administrator-password`** command can be removed. Then, the configuration can be pasted back into the switch.

6.8 Other considerations

There are a few other switch features that are preferred when you secure the switch. The first is the default inactivity timeout setting, which is 5 minutes. This value is a good value to keep unless local security policies recommend a different value.

Also available is a display pre-login notice to warn about unauthorized access. Most companies have standard notices that should be used. Figure 6-1 on page 123 shows a command to create a pre-login notice.

```
enable
configure terminal
system notice
*************************************************************************
*                                                                       *
*  NOTICE:  Access to this device is restricted to authorized users for *
*           business purposes only. Unauthorized access is not allowed  *
*           and will be prosecuted to the fullest extent of the law.    *
*                                                                       *
*           This system may be monitored for administrative and security *
*           reasons. By proceeding, you acknowledge that you have read  *
*           and understand this notice and you consent to the system    *
*           monitoring.                                                 *
*                                                                       *
*************************************************************************
.
```

Figure 6-1 Command to create a pre-login notice

Note: The period on the last line is important because it is required to end the notice entry command.

7

Operation and management

This chapter describes preferred practices for the operation and management of Lenovo Networking switches during initial deployment and their ongoing use in a production environment.

The chapter includes the following topics:

7.1 Initial deployment practices and considerations

In this section, preferred practices for the initial deployment of Lenovo switches are reviewed. The objective of these practices is to integrate the switches into the customer environment without causing a disruption to that environment.

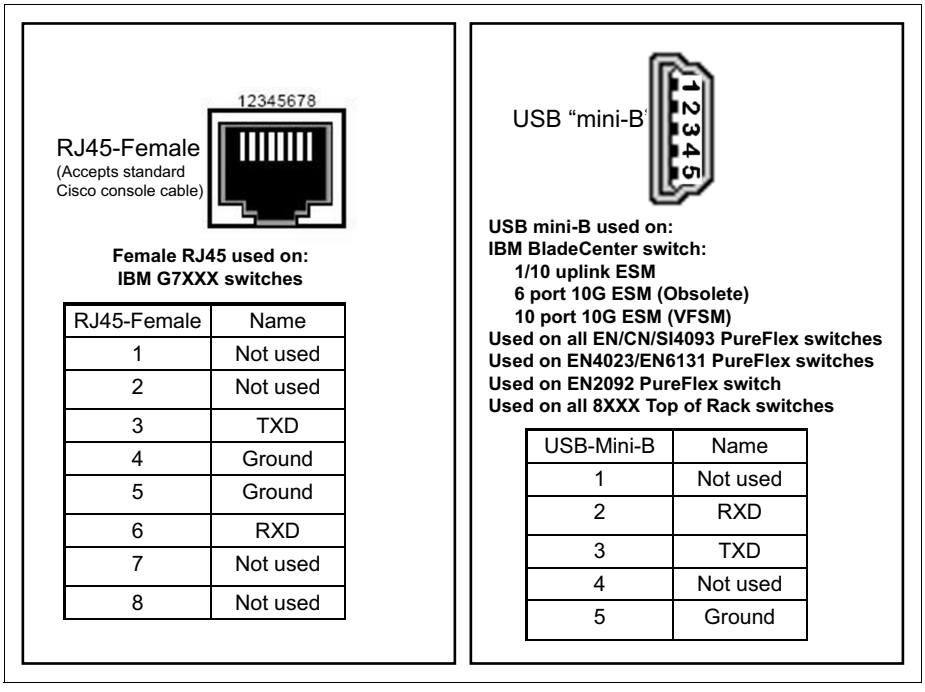
7.1.1 Console access

All of the Lenovo switch product offerings include a serial console port, which in the case of currently shipping products uses a mini-B-USB connector or a RJ45 connector (on the G7*XXX* top of rack [ToR] switches only). Some older products use a full-size USB-A type connector. In all cases, these connectors use RS-232 serial signaling and are intended to be used with the console cable that is orderable or included with the switches.

Console cables are passive and the pinout for constructing some of these cables is shown in Figure 7-1, Figure 7-2 on page 127, and Figure 7-3 on page 127. Connecting a true USB adapter (or any other USB device) directly to the USB-style serial console ports (without first going through a serial to USB adapter) is not supported and can damage the port.

The default settings for all Lenovo serial console ports are 9600 baud, N, 8, 1 and can be changed to other settings if wanted.

Figure 7-1 shows the port pinouts for the RJ45 style console port on the G7*XXX* series of switches and the Mini-B-USB console style port on most other embedded and ToR switches.

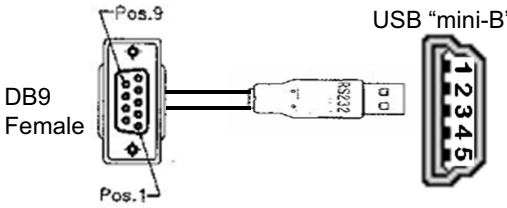


RJ45-Female	Name
1	Not used
2	Not used
3	TXD
4	Ground
5	Ground
6	RXD
7	Not used
8	Not used

USB-Mini-B	Name
1	Not used
2	RXD
3	TXD
4	Not used
5	Ground

Figure 7-1 Cable pin-out for serial RJ45 and USB Mini-B console ports

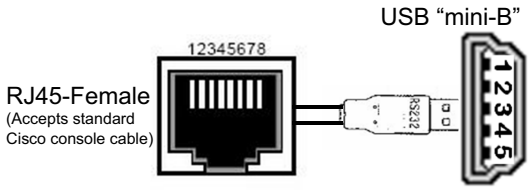
Figure 7-2 shows a cable pin-out for a serial mini-B-USB to DB9 console cable that is shipped with some switches.



DB9	Name	Wire Flow	Comment	Name	USB
5	Common			Ground	5
3	TD			Data -	2
2	RD			Data +	3
N/A	N/A	N/A	(not connected)	+5V	1
N/A	N/A	N/A	(not connected)	ID	4
1	DCD		Pins 1, 4 and 6 looped on DB9 end		
4	DTR				
6	DSR				
7	RTS		Pins 7, 8 looped on DB9 end		
8	CTS				

Figure 7-2 Cable pin-out for serial USB-mini-B to DB9 console cable

Figure 7-3 shows a cable pin-out for a serial USB-mini-B to female RJ45 console cable that can be built to connect to a typical Cisco console cable.



RJ45-Female	Name	Wire Flow	Comment	Name	USB
4	Ground		(not connected)	ID	4
5	Ground		Pin 4 connected to Pin 5 on back of RJ45	Ground	5
6	RXD			Data -	2
3	TXD			Data +	3
N/A	N/A	N/A	(not connected)	+5V	1
1			Pin 1 connected to Pin 8 on back of RJ45 (to loop RTS and CTS)		
8					
2	DTR		Pin 2 connected to Pin 7 on back of RJ45		
7	DSR				

Figure 7-3 Cable pin-out for serial USB-mini-B to RJ45 (Cisco style) console cable

Figure 7-4 shows ports on an EN4093 switch, with the serial port on the right.

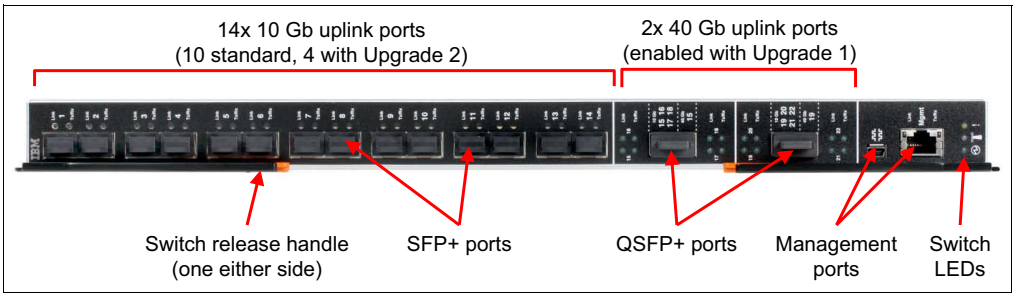


Figure 7-4 Ports on an EN4093 switch, with the serial port near the bottom

Figure 7-5 shows a G8264 switch front with a console port at the lower left.

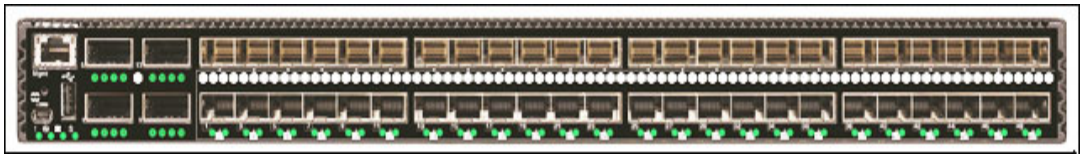


Figure 7-5 G8264 switch front with a console port at the lower left

One important aspect of the use of the serial console port is the viewing of low-level messages during the boot process that are not syslog messages and are not viewable in any other way because IP-based access is not available until the boot process is substantially complete. There are also some low-level error messages that can be sent out of the serial console port but not sent via other methods.

Shift+B console boot menu

A menu is accessible via the serial console during the boot process by pressing **Shift+B** while dots (periods) are displayed during the initial memory test. By using this menu, you can select the firmware image (image1 or image2), which saved configuration to use (primary, backup or factory default), and use X-modem to upload a firmware image. The X-modem upload is intended as a last effort before you send a switch back for service. It often is used when another upload technique fails in midstream, such as because of a power failure.

For more information about the functions of the menu, see the *Installation Guide* for the various products.

Figure 7-6 shows this menu and examples of changing the boot image and config file.

```
Boot Management Menu
   1 - Change booting image
   2 - Change configuration block
   3 - Boot in recovery mode (tftp and xmodem download of images to recover
switch)
   4 - Xmodem download (for boot image only - use recovery mode for application
images)
   5 - Reboot
   6 - Exit
Please choose your menu option: 1

Enter image to boot: 1 or 2 (currently set to 1): 1
Booting from image 1

Boot Management Menu
   1 - Change booting image
   2 - Change configuration block
   3 - Boot in recovery mode (tftp and xmodem download of images to recover
switch)
   4 - Xmodem download (for boot image only - use recovery mode for application
images)
   5 - Reboot
   6 - Exit
Please choose your menu option: 2

Currently using active configuration block
Enter configuration block: a, b or f (active, backup or factory): a
```

Figure 7-6 Shift-B boot menu; accessible from serial console during early stages of the boot process

7.1.2 Default IP addresses

On embedded switches, the primary management IP address is learned from the advanced management module (AMM) or chassis management module (CMM) at boot time and must be configured on the management module (MM). If not configured, the default addresses that are used (as assigned by the AMM (for BladeCenter) or CMM (for Flex System) are as shown in Table 7-1.

Table 7-1 Default IPs for embedded switches and management modules

Module location	Default IP: BladeCenter	Default IP: Flex System
AMM or CMM default IP	192.168.70.125/24	192.168.70/100/24
Switch bay 1	192.168.70.127	192.168.70.120
Switch bay 2	192.168.70.128	192.168.70.121
Switch bay 3	192.168.70.129	192.168.70.122
Switch bay 4	192.168.70.130	192.168.70.123
Switch bay 5	192.168.70.131	N/A
Switch bay 6	192.168.70.132	N/A

Module location	Default IP: BladeCenter	Default IP: Flex System
Switch bay 7	192.168.70.133	N/A
Switch bay 8	192.168.70.134	N/A
Switch bay 9	192.168.70.135	N/A
Switch bay 10	192.168.70.136	N/A

The primary rule for assigning IP addresses for embedded switches from the associated Management Module is that they must be assigned in the same IP subnet that is used by the Management Module, and are associated with interface IP 128 on the switch. It is also possible to configure and use IP addresses on embedded switches that make use of an available dedicated Management port (for example, EXTM on the 4093 series of Flex System switches that are associated with interface IP 127) or via inband data ports that are associated with interface IP 1 - 124).

Stand-alone, top of rack switches can learn their default management IP address with DHCP or BootP and have the default addresses as listed in Table 7-2 if not learned via DHCP/BootP or manually assigned.

Table 7-2 Default IPs for Top of Rack switches

Switch Type	Default IP
Top of rack - Management Port	192.168.50.50/24
Top of rack - Data ports (VLAN 1)	192.168.49.50/24

7.1.3 Preferred practices for a nondisruptive installation

The following practices are designed to prevent errors and issues during an installation. In some cases, these issues can cause serious disruption to a network:

- Do not allow the switch to be connected to the network until it is configured. This situation can be avoided by not connecting the cables until the configuration is loaded, by disabling the ports on the upstream switches that connect to the new switches, or by disabling the uplink ports on the new switch. The reason for this process is to avoid a network loop when the new switch is connected without configuration of VLANs and spanning tree. On some products, ports default to a *blackhole VLAN* on an unconfigured switch, and all traffic that is received on ports in such a VLAN is discarded. However, it is a good practice to not have active connections while performing initial configurations.
- When the switch is configured and booted, check that you are running the intended version of firmware and upgrade it as needed.
- After you verify the configuration, ensure that it is saved to flash memory (use the **`copy run start`** command, and ensure that you answer 'y' when you are asked if the newly saved configuration is to be used at the next reboot). If this step is not performed, you get a factory default configuration at reboot, whether planned or because of a power failure or a similar situation. To verify that the saved configuration is used on the next reload rather than the factory default, use the **`show boot`** command.
- Limit the use of VLAN 1. On products that do not implement the blackhole VLAN, ports default to VLAN 1. It is a good practice to not use VLAN 1 for any production traffic, and to remove unused ports from VLAN 1. Many Cisco products also set unused ports to VLAN 1 in the default settings.

- Leave unused ports disabled. This setting prevents unexpected or unauthorized connections to these ports. It is a good practice from a security perspective and to prevent erroneous connections.
- Use caution when you are connecting ports that use static aggregation. One quick method to create a network loop and a broadcast storm is to have two parallel links between two devices that are aggregated on one device but not on the other. This situation can happen if a cable is plugged into the wrong port on one side.

  Complete the following steps to avoid errors for a two-port aggregation:

  a. Connect both cables to the switch on one side and enable the ports.
  b. Connect one cable to the other switch and verify that the link comes up on both sides.
  c. Use Link Layer Discovery Protocol (LLDP) or another method (see 7.10, “LLDP recommendations” on page 150) to verify that the cable is connected to the correct port on both sides.
  d. Disable the port that you connected to the second switch in Step b.
  e. Plug in the other cable and repeat the verification process.
  f. Enable the port that you disabled in Step d.

7.2 Basic configuration

Basic standard configuration templates are typical in most environments. Some examples of these common and useful commands are described in this section.

7.2.1 System Notice (pre-login notice)

The **system notice** command specifies that a notice to be displayed before the “Enter password:” prompt. This notice can contain up to 1024 characters and new lines.

Example 7-1 shows a login notice that is presented to a user via the command-line interface (CLI) before a login attempt. The use of this format is a prompted entry and not suitable for cutting and pasting.

Example 7-1 Adding system notice in an interactive prompted fashion

```
config t
system notice
! Will see a message such as “Enter new login notice line by line (enter single
! '.' to end) :”
All access to XYZ corporate information system resources is restricted
to authorized users. Users must uniquely identify themselves using their
personal credentials, and will be held accountable for all activities under
this identification. UNAUTHORIZED ACCESS OR MODIFICATION TO THIS SYSTEM IS
PROHIBITED AND MAY BE A CRIMINAL OFFENSE. XYZ WILL REFER SUCH ACTIVITY
TO LAW ENFORCEMENT AUTHORITIES AND OFFENDERS MAY BE SUBJECT TO CRIMINAL
PROSECUTION AND/OR CIVIL ACTION.
! Enter a single period and press the enter key to terminate the entry
.
```

This same message also can be entered by using the **system notice addline** command for each line. It also permits cutting and pasting in the system notice, as shown in Example 7-2 on page 132.

Example 7-2 System notice suitable for cutting and pasting (lines are wrapped)

```
conf t
system notice addline "All access to XYZ corporate information system resources is
restricted"
system notice addline "to authorized users. Users must uniquely identify
themselves using their"
system notice addline "personal credentials, and will be held accountable for all
activities under"
system notice addline "this identification. UNAUTHORIZED ACCESS OR MODIFICATION TO
THIS SYSTEM IS"
system notice addline "PROHIBITED AND MAY BE A CRIMINAL OFFENSE. XYZ WILL REFER
SUCH ACTIVITY"
system notice addline "TO LAW ENFORCEMENT AUTHORITIES AND OFFENDERS MAY BE SUBJECT
TO CRIMINAL"
system notice addline "PROSECUTION AND/OR CIVIL ACTION."
```

7.2.2 Banner

When a user or administrator logs in to a switch, the login banner is displayed as part of the welcome window. A banner can include 1 - 80 characters.

Example 7-3 shows a login banner that is presented to a user via the CLI after a successful login attempt.

Example 7-3 Example of a banner

```
config t
banner "Unauthorized access to this device is prohibited!"
```

7.2.3 Logging (Syslog) server

You can use system logging to collect and store switch log activity locally and to a syslog server. Up to 2000 syslog messages can be stored locally (not configurable) on the switch, and up to two remote syslog servers can be configured to receive this same log data.

Example 7-4 shows an example of configuring two syslog servers. The preferred option is to select the `mgt-port`, which represents the 10/100/1000 port for out of band management on ToR switches, or the CMM facing port on Flex System embedded switches. Optionally, the inband `data-port` can also be used as the source for ToR and embedded switches, and the option of `extm-port` for Flex System out-of-band EXTM RJ45 port.

Example 7-4 Logging server 1 and 2 configuration example for ToR or Flex System embedded switch

```
logging host 1 address 10.10.10.1 mgt
logging host 2 address 172.16.20.2 data
```

The syntax is the same for BladeCenter embedded switches, without the trailing mgt, data, or extm option; for example, **`logging host 1 address 10.10.10.1`**.

System logging allows for two options of levels that can be defined, *Severity* and *Facility*. Severity sets the severity level of host 1 or 2. The default level is 7, which means to log all severity levels.

Table 7-3 lists the available severity levels.

Table 7-3 List of available severity levels

Level	Description
0	Emergency: System is unusable
1	Alert: Action must be taken immediately
2	Critical: Critical conditions
3	Error: Error conditions
4	Warning: Warning conditions
5	Notice: Normal but significant condition
6	Informational: Informational messages
7	Debug: Debug-level messages

Facility sets the facility level of host 1 or 2. The default is 0, which means to log against the kernel messages only. The Facility value is a way of determining which process on the system created the message.

Table 7-4 lists the available facility levels.

Table 7-4 List of available facilities levels

Level	Description
0	Kernel messages
1	User-level messages
2	Mail system
3	System daemons
4	Security/authorization messages
5	Messages generated internally by syslog
6	Line printer subsystem
7	Network news subsystem

By using the **`logging source-interface`** command, you can set a loopback interface (1 - 5) as the source IP address for logs.

By using the **`logging console`** command (enabled by default), logs to be seen on a CLI session and sent to any configured logging hosts. To disable it, use the **`no logging console`** command. Disabling logging via the **`no logging console`** command affects only the messages that are sent to the CLI, not those messages that are sent to any configured logging hosts.

7.2.4 Host name

The switch host name is configured by using the **hostname** command, and is used as the CLI prompt going forward and in LLDP messages to help identify this device from other devices. When you change the host name, it also changes the Simple Network Management Protocol (SNMP) name of that device.

7.2.5 System idle (CLI timeout)

Sets the **system idle** timeout command for CLI sessions in minutes. The default value is 10 minutes, and the supported range is 0 - 60. Values of 1 - 60 represent the number of minutes of inactivity before the connection times out. A value of 0 disables system idle time out (never time out).

7.2.6 Terminal Length (per session)

By using the **terminal length** command, a user can change the number of lines per window. Terminal Length lasts on a per-user session only. After the user logs out, the length reverts to VTY or Console (for more information, see “Line VTY length (configuration change, telnet/ssh)” and “Line console length (configuration change, console)”). A value of 0 disables paging. That is, it displays the full output without pause. A minimum of 0 and maximum of 300 are supported. The default is 28 lines per page.

7.2.7 Line VTY length (configuration change, telnet/ssh)

By using the **line vty** length command, a user can change the number of lines per window for Telnet and SSH sessions. Line VTY is a configurable option, but requires a logout and login for the configuration change to take effect. A value of 0 disables paging (no pause). A minimum of 0 and maximum of 300 are supported. The default is 28 lines per page.

7.2.8 Line console length (configuration change, console)

By using the **line console** length command, a user can change the number of lines per window for serial console sessions. Line console is a configurable option; however, as with Line VTY, it requires a logout and login for the configuration change to take effect. A value of 0 disables paging (no pause). A minimum of 0 and maximum of 300 are supported. The default setting is 28 lines per page.

7.2.9 Changing CLI modes

By using the **cli-mode** command, a user can choose between the older ibmnos-cli, a more Linux like CLI, and a newer iscli mode, which is a more industry standard CLI that is also used on other vendor switches.

The **cli-mode** command also offers the option to enable **prompt,** with which the user can select between the two modes of CLI operation at the time of login. When the prompted mode is used, after a user is logged on and selects a CLI mode, any subsequent users are automatically logged in by using the current active CLI mode that is selected by that current user (without the option to choose a CLI mode).

Note: CLI modes cannot be changed for some later versions of software on products with ibm-nos. In these cases, the only CLI is iscli.

7.2.10 Preferred practices for initial installation

At initial configuration, complete the following steps for all switches:

1. Predefine enabled VLANs and add them to the appropriate ports.
2. Configure interoperable Spanning Tree Protocol (STP), including, but not limited to, bridge priority, portfast and edge, bpdu-guard, and loop-guard.
3. Configure all required LACP or static PortChannel interfaces.
4. Set up any wanted host name, logging servers, Network Time Protocol (NTP), and so on.
5. Turn off all unused ports (turn them on only when both sides of that link are fully configured and validated).

Complete these steps before cabling the switch or bringing up any of the ports.

Tip: A correct configuration for ports, PortChannels and Layer 2 features (for example, Spanning Tree and VLANs), and Layer 3 that includes static and dynamic routes often can be set up after you connect the switches to the network.

7.3 Operational command considerations

Some commands are considered to be *operational* because running the command can affect operation of the switch but does not change the running configuration of the switch. Unlike commands that are placed into the running configuration, operational commands do not need to be run from **`config t`** mode. These operational commands are often referred to as *oper* commands.

Some of these oper commands are industry standard, such as clearing various counters and tables (for more information, see 7.4, “Clearing tables and counters” on page 136). Some commands are a bit different from commands that might be found in other vendors’ products. One such example is the ability to shut down and start interfaces without going into the **`config t`** mode, as shown in Example 7-5.

Example 7-5 Sample commands for shutting down ports and start ports by using the oper format

```
interface port <port number or alias> shutdown
no interface port <port number or alias> shutdown
```

A **`show int status`** command that is run when a port is operationally disabled (with commands such as the commands that are shown in Example 7-5), shows only that the port is **`disabled`**, but provides no information about *why* it is disabled. (The **`shutdown`** is not shown in a **`show run`** command.) To see the operational status for a port, run the **`show interface port <port number or alias> operation`** command, as shown in Example 7-6 on page 136.

Example 7-6 The use of an oper shutdown command and checking status

```
G8264T-2#int port 17 shutdown
Port 17 disabled.
Feb 10 12:20:30 G8264T-2 NOTICE  link: link down on port 17
G8264T-2#show int port 17 oper
Current Port 17 operational state: disabled, FDB Learning ON
G8264T-2#
```

The following example of an oper command can force a VRRP master to become a backup:

```
router vrrp backup <virtual router number (1-255)>
```

For more information about the available oper commands for a specific release of software and model of switch, see the respective *Command Reference* for the product. Most Command References for Lenovo switches include a chapter (often referred to as *Operations Commands* chapter) about the available oper commands.

7.4 Clearing tables and counters

Manipulating the various tables and counters of a switch can be important for supporting any environment. Lenovo switches have many such tables and counters for many purposes. A good way to see items that can be cleared is to use the **clear ?** command.

Example 7-7 shows a listing of available counters and tables that can be cleared on a G8264CS switch (options can vary between models and versions of firmware).

Example 7-7 High-level clear commands

```
G8264CS#clear ?
Reset functions
  access              Clear network access to this system
  access-control      Clear access control statistics
  arp                 Clear ARP cache
  counters            Clears all interfaces statistics
  cpu                 Clear CPU Utilization
  fcoe                Clear FCOE features
  flash-config        Clear all flash configurations
  flash-dump          Clear flash dump
  hotlinks            Clear Hot Links statistics
  ike-sa              Clear IKEv2 SA
  interface           Select an interface
  interfaces          Clears all interfaces statistics
  ip                  Clear IP statistics
  ipsec-sa            Clear IPsec SA
  ipv6                Clear IP6 statistics
  line                Line information
  lldp                Clear LLDP remote devices information
  logging             Clear syslog messages
  mac-address-table   Clear MAC forwarding table
  mp                  Clear all CPU packet statistics and logs
  mp-counters         Clear all MP related stats
  ntp                 Clear NTP statistics
  qos                 Clear quality of service counters
  snmp                Clear snmp statistics
```

```
  ssh-clienthostkey  Clear SFTP host keys from the client database
  statistics         Clear statistics
  virt               Clear Virtual machine statistics
  vlag               Clear vLAG
  zone               Remove Zone Database
```

Some clear commands have more options beneath the level that is shown in Example 7-7 on page 136. Example 7-8 shows the options for the **clear access-control** command.

Example 7-8 Clear options below the access-control level

```
8264CS-Top#clear access-control ?
  list   Clear access control list statistics
  list6  Clear IPv6 access control list statistics
  meter  Clear access control metering statistics
  vmap   Clear vlan map statistics
```

The following list shows some common tables and how to clear and view them (any **X** that is listed in a command represents a port or a range of ports that is separated by commas or dashes):

- **clear interfaces** or **clear counters** clear all interface counters on all ports. You also run the **clear interface X** command to clear only a single interface or range of interfaces. The following interface counters are available. Use the **show int port X ?** command to see all available options:
  - Use the **show int port X interface-counters** command to see a snapshot of interface input/output counters.
  - Use the **show int port X maintenance-counters** command to see a detailed listing of counters for these interfaces.
  - Use the **show int port X lacp counters** command to see a detailed listing of LACPDU counters for these interfaces.
  - Use the **show int port X bitrate-usage** command to see a 1-second, self-updating snapshot of input and output rates for these interfaces.
- **clear mac-address** clears the current MAC table. Use the **show mac-address-table** command to dump the current MAC tables

  There are many more MAC-table-related counters. This list provides just a few examples. Use the **show mac-address-table ?** command to display all available options

7.5 Firmware upgrade considerations

The procedures for upgrading code on a Lenovo switch are provided in various places, such as release notes or installation guides.

Code can be downloaded to the switch with the CLI and the browser-based interface (GUI), and over any dedicated management ports, data ports, or serial console port.

Code can be downloaded to the switch with TFTP and HTTP, in addition to FTP and SFTP on most recent software.

When the isCLI mode is used, use the **copy** command to load code on the switch; for example: **copy tftp image1**. This command prompts you to enter the TFTP server and image name information and then downloads that code to the image1 slot.

It is also possible to put the entire command with all needed information to complete the download on a single line (no interactive prompting is needed) to provide a single-line command to upgrade the image or boot slots. The following example shows copying an operating system to the image 1 slot from a TFTP server at 1.1.1.11 with the G8264 out-of-band management port:

```
copy tftp image1 address 1.1.1.11 filename G8264-CS-7.8.7.0_OS.img mgt-port
```

All code is checked to ensure that the following elements are correct before it is written to flash:

- It is the correct code for this switch model. If the code is not for this model switch, it is rejected.
- It is the correct code for this slot (boot or operating system). You cannot put boot code in an operating system image slot, or vice versa.
- The CRC is correct (the code is not corrupted). the switch does not write the file to flash if the computed CRC does not match the stored CRC of the file that is downloaded.

If you are updating old code (multiple major releases difference), it is necessary to first update to an interim release in some cases. Be sure to review the release notes for interim requirements. A common indicator that an interim release is needed is a message about insufficient space on flash when you attempt to download the code to the switch.

Do not be concerned about the amount of flash space on a switch because no code is released that does not fit into the flash for the switch. An insufficient space message is used to indicate that an interim release is needed to reformat the flash area for a larger image for which was previously allotted. That interim release fixes the flash size message.

The remainder of this section describes attributes of the process that might not be well-documented in other locations.

Regarding Lenovo converged Ethernet switches, such as the RackSwitch 8264CS and Flex System CN4093, these devices have separate ASICs for Ethernet and Fibre Channel functionality. The firmware images for these switches contain an image for each ASIC. If FCoE is used on the switch, ensure that the switch configuration already has Converged Enhanced Ethernet (CEE) enabled by using the **`cee enable`** command before the firmware is upgraded. This check ensures that the Fibre Channel engine of the switch also is properly updated during the firmware flash.

If CEE is enabled later, it is recommended that the current image is reflashed to ensure that the Fibre Channel code was correctly updated. To verify the versions of firmware that are installed on the Ethernet, run the **`show version brief`** command. To verify the versions of firmware that are installed on the Fibre Channel ASICs, run the **`show fc-internals`** command.

Currently, all Lenovo switches use two code files to start a switch: a boot file and an operating system file. All Lenovo switches have a single flash location for the boot file (called `boot`), and two locations to store operating system files (called `image1` and `image2`). The user can select which image location from which to boot the operating system. To see the images that are installed in flash and other important boot-time attributes, use the **`show boot`** command. Example 7-9 on page 139 shows the output of a **`show boot`** command as taken from a G8264CS switch.

Example 7-9 Output of show boot command from a G8264CS

```
8264CS-Top#show boot
Current running image version:7.8.6
Currently set to boot software image1, active config block.
NetBoot: disabled, NetBoot tftp server:  , NetBoot cfgfile:
USB Boot: disabled
Current CLI mode set to ISCLI with selectable prompt disabled.
Current FLASH software:
  image1: version 7.8.6, downloaded 21:31:03 Sun Feb 10, 2001, Mode Stand-alone
  image2: version 7.1.2, downloaded 18:27:57 Sun Jan  1, 2000, Mode Stand-alone
  boot kernel: version 7.8.6, Mode Stand-alone
Currently scheduled reboot time: none

Copy Mode: Stand-alone
```

The following output of the **show boot** command is shown in Example 7-9:

- `Current running image version:7.8.6` represents the operating system image that was used in the last reload. This image can also be displayed by using the **show version** or **show version brief** commands.
- `Currently set to boot software image1, active config block` provides boot information. The first part shows that, for the next reload, this switch uses the operating system file in the `image1` slot. This selection can be changed by using the **boot image imageX** command. The second part (`active config block`) shows that on the next reload, the switch uses a configuration that is saved in NVRAM.

  The fact that a config is saved to NVRAM does not mean that it is used on a reload. This setting must be changed to `active` if you want to use the saved config on the next reload (use the **boot conf active** command). When you run the **copy running startup** (or **write mem**) command, it saves the running config to NVRAM and then prompts you **y/n** to change this setting to the active and startup config if the switch is set to use the factory default config.
- `NetBoot: disabled, NetBoot tftp server:  , NetBoot cfgfile:` shows that netboot is not being used. If the netboot feature is used, these values are populated.
- `USB Boot: disabled` shows that by default, the switches boot from the images that are stored locally in flash. They can be instructed to boot from an image on a USB flash drive that is plugged into the switch. Not all Lenovo switches have USB ports for this purpose.
- `Current CLI mode set to ISCLI with selectable prompt disabled` describes the CLI mode settings. Most Lenovo switches support two different CLI modes. The default setting for top-of-rack switches is the isCLI mode (industry-standard CLI with commands that might be similar to Cisco commands). There is also a menu-driven CLI that, usually is being phased out. However, it is still the default setting on certain switches (for all BladeCenter switches and older Flex System switch software, the default setting is this menu driven CLI). It is also possible to tell the switch to prompt a user what CLI to use when logging in.

  Based on this output, this switch goes into isCLI mode upon login without prompting to use a different CLI. Use the **boot cli-mode <options>** isCLI command to change these values. Changing CLI modes requires a reload to take effect. Changing to prompted mode can be done without a reload. Lastly, if a switch is in prompted mode and a user is already logged in to a CLI, they are not offered a prompt. Instead, the switch goes directly into the CLI mode the other user is already using (the switch does not allow two users to log in at the same time with different CLI modes).

In later software, some switches have no option for the menu-driven CLI. For these versions, all references to selecting the CLI mode are removed, including the **boot cli-mode** command. For switches that have isCLI available only, the output of the **show boot** command does not include information about the CLI that is in use.

- `Current FLASH software:`
  `image1: version 7.8.6, downloaded 21:31:03 Sun Feb 10, 2001, Mode Stand-alone`
  `image2: version 7.1.2, downloaded 18:27:57 Sun Jan  1, 2000, Mode Stand-alone`
  `boot kernel: version 7.8.6, Mode Stand-alone`

  This extract shows what versions of code are in the various flash locations. The `Mode Stand-alone` information is specific to the G8264CS switch and its support of Flex System Interconnect Fabric (FSIF) mode. In this example output, the switch is in stand-alone mode and not in FSIF mode.

  Although two images are stored, only one ever is used at a time during the boot process. The other is not used unless it is selected and then reloaded to use it.

- `Currently scheduled reboot time: none` shows when the switch is set to reload. Lenovo switches can be set to reload at a certain date and time (for example, to perform an unattended reload after hours to boot into new code). Use the **boot schedule** command to set a timed reload.
- `Copy Mode: Stand-alone` is also specific to the G8264CS and FSIF switches. It is used when you are changing between FSIF and non-FSIF modes to support copying the necessary image files that are needed in FSIF mode (that contain multiple images). This setting can be changed by using the **copy** command.

It is a preferred practice to always keep whatever operating system image is booting at the same level as the `boot` file. That is, if `image1` has 7.7.1 and is selected as the image to use for the next reload, ensure that the boot file also contains version 7.7.1. In the past, when the **boot** code was more static (not changing from version to version), a user might store and choose the operating system code to load in `image1` or `image2` (for example, `image1` might be the latest code, but `image2` might be older code to be used as a fallback).

If revisions are not too far apart in versions, newer boot files can often be used for older versions of the operating system (`imageX`) files. However, to ensure that issues are not encountered, keep the selected image operating system file in sync with the boot file version.

During fallback, some or all of a config might be lost (some commands in newer code might not be present, or might have a different syntax in older code). Differences are more likely to occur for larger differences between the running version and the older version that is about to be booted. *Always* save the config to a file before you fall back to an earlier version.

7.6 Configuration control considerations

Configuration files on Lenovo switches are stored in preset locations, with one location referred to as *startup* (also known as *active*) and one called *backup*. These locations are strictly controlled by the Lenovo CLI, which makes this process slightly different from how Cisco performs this process (Cisco switches use a file that is stored on a file system, which can be manipulated as a file).

It is important to understand how Lenovo switches use these locations in NVRAM to ensure that configs are saved and used properly. Before you look at how Lenovo switches store and work with config files, it is helpful to show how a typical Cisco switch does these tasks for comparison. Figure 7-7 shows the sequence of events that occur when certain commands are run on a Cisco switch.

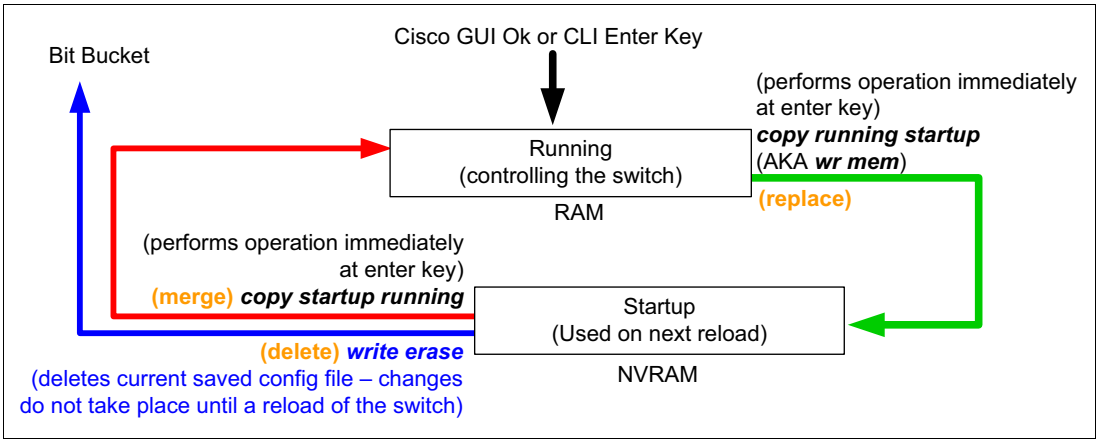


Figure 7-7 Cisco configuration operations

Figure 7-8 shows the Lenovo NVRAM locations for configurations and how the Lenovo switch operates on these areas.

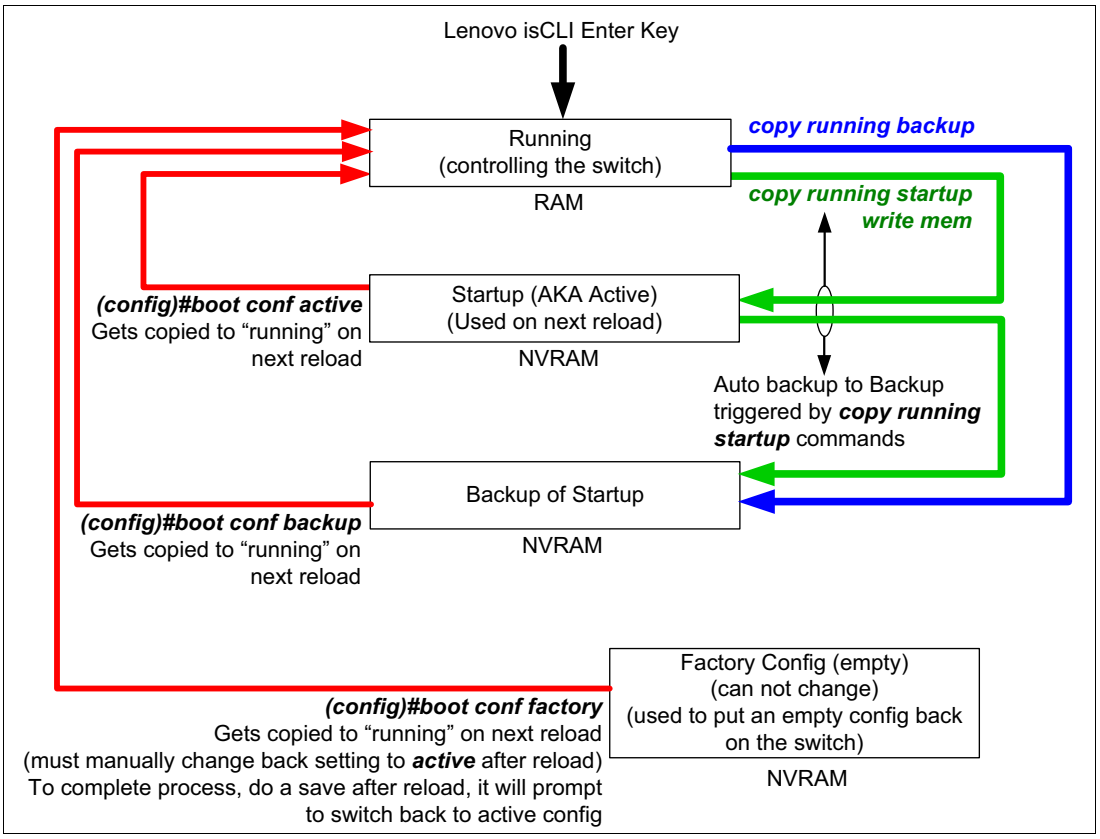


Figure 7-8 Lenovo isCLI configuration operations

Consider the following points regarding Figure 7-8 on page 141:

- The Lenovo example that is shown is specific to the Lenovo isCLI mode. The Lenovo menu-driven CLI uses a different approach, and is not described in this section.
- The Lenovo switches keep a running backup of the last saved config. When you run the **`copy running startup`** command (or **`write mem`**) command, the switch first copies the saved config in the *startup* location to the *backup* location, and only then does it copy the configuration that is in running memory to the *startup* location. This process occurs automatically when these commands are run.

With the backup config being a backup to the startup config, a user can go back one saved level and retrieve a previous config if necessary by using a **`show backup`** command and copying out the information. The user also can set the next boot to boot directly from the backup area of NVRAM (by using the **`boot config backup`** command). consider the following points:

- It also is possible to copy the *running* config directly into the *backup* location (by using the **`copy running backup`** command).
- When you return a switch to the factory default config (by using the **`boot config factory`** command), the startup and backup configs are not deleted. Instead, the pointer from where to boot is changed to point to a special factory default area of NVRAM. To erase the startup and backup areas of NVRAM (instead of changing the pointer to the empty factory default block), use the **`clear flash-config`** command.

When you are working with a Lenovo switch that is set with the pointer to use the factory default block, there is a prompt that is similar to the prompt that is shown in Example 7-10 when you save a config.

Example 7-10 Prompt that is shown when the factory default block is in use

```
Switch is set to use factory default config block on next boot.
Do you want to change that to the active config block (y/n) ?
```

If you answer **`no`** to this prompt, the config is still saved to NVRAM, but the pointer remains pointing at the empty factory default location. This request message is sometimes missed when other messages open and the change to the pointer is lost. To ensure that the pointer is correct, use the **`show boot`** command to confirm that the active config block is used on the next reload.

7.7 Embedded switch considerations

Lenovo embedded switches (installed in Flex System and BladeCenter chassis) are true switches and do all of the normal tasks that are expected of this class of networking device. However, because they are part of an enclosure, they have some unique attributes. This section describes some of these unique elements and potential areas of confusion.

7.7.1 Common to Flex System and BladeCenter switches

Flex System and BladeCenter enclosures support several switch models from Lenovo and third parties. These enclosures include a special chassis management module, which is the MM or AMM in BladeCenter, and the CMM in the Flex System chassis. These management modules provide a single point of management path for all components that are installed in the enclosure, including the switch modules.

Aside from this management path via the chassis management module, embedded switches are essentially stand-alone switches that are packaged for use in an enclosure-based environment. It is sometimes assumed that a pair of embedded switches provides some built-in method to back each other up; however, this configuration is not the case. The switches can (and often are) deployed in redundant pairs. However, it is the configurations on the switches, hosts, and upstream network that make them redundant (as with stand-alone switches), not some special feature or connections that are built into the enclosure.

The presence of the management module and its associated uplink path for switch management is one area where enclosure switches often vary from stand-alone switches.

Note: It is critical that the CMM, IMM, and the management interfaces for switches within the Flex System chassis are not placed in a data VLAN/subnet that is used by hosts that are installed inside that chassis. The CMM uses IP forwarding and proxy ARP functionality to provide network access to internal chassis management interfaces. If operating systems that are running on hosts that are installed in the Flex System chassis are on the same network, the CMM might inadvertently respond to ARPs by the operating system, which leads to the operating system losing connectivity to the rest of the network.

Because of the potential conflict and loss of operating system connectivity, a configuration that places host operating systems on the same VLAN/subnet as an AMM, CMM, IMM, or switch management interface is not supported.

The following helpful tips and examples show how embedded switch configuration differs from stand-alone switches and the potential affect to the administrator:

- The following items can be controlled on the switch by the CMM:
  - Set the IP address or mask and default gateway on the switch that is to be used by the switch via the MM uplinks.

    This information is defined and saved on the MM and sent over via the I2C serial bus between the management module and the switch. To work properly, this IP information must be defined in the same IP subnet that the MM uses on its own uplink.
  - Reset the switch to the factory default settings.
  - Override local control of uplink settings and force the uplinks into a shut state.

    On Flex System switches, the default setting enables the switch to control its own uplinks.

    On BladeCenter switches, the default setting is to hold the ports in a shut state until they are enabled at least one time for the switch bay from the MM.
  - Reload the switch and turn it on and off.
  - Turn off the switch if the MM detects over temp or over current related conditions for the switch bay
  - Allow or disallow management connections to the switch via the uplinks of the switch (on a separate IP interface from the one provided by the MM).

    The default setting is to disable management via the uplinks on Flex System and BladeCenter based switches.
- When you log in to an enclosure-based switch, it is helpful to know exactly in which bay and chassis the switch is installed. A useful command for enclosure-based switches is the **`show system chassis`** command. When this command is used, several important pieces of information are returned, including the following information:
  - The bay number in which the switch is installed.

  - The Universally Unique Identifier (UUID), which can be used to determine what chassis the switch is installed.

- The MM provides an IP address and default gateway to the switches that is controlled by the MM only and cannot be changed from the switches. This IP information is for switch management over the uplink as the MM to permit external control sessions (for example, telnet and SSH) to the switches via the management module uplink. To see this MM that is assigned IP information, use the **`show ip info`** command and look for IP interface 128 and default gateway 4 (which are strictly assigned and controlled by the MM).
- Embedded switches can also be managed over their own external uplinks; however, in the default settings, this management is disabled. If a user wants to use an IP interface via the uplinks (or the EXTM management port in the case of the 4093 series switches), this feature *must* be enabled from the MM before it is permitted. If this feature is not enabled, a user can still ping the inband or EXTM assigned IP address, but any attempt to connect to it (for example, telnet or SSH) is rejected. Consider the following points:
  - It is possible to tell if this feature is enabled on the MM by running the **`show system chassis`** command on the switch. For more information about how to use this output to determine whether the CMM is allowing management of the switches via their own uplinks, see “Viewing MM, AMM, or CMM controlled features from the switch side” on page 145.
  - It might appear as though there is a physical path between the uplinks of the enclosure switch and the CMM that permits managing the MM via the switch uplinks. However, it is not possible to manage the MM via the switch uplinks. The CMM can be managed by its own external uplink only.
- It is not possible to ping between switch modules within the same enclosure by using the MM path (although all switches in an enclosure are usually on the same IP subnet). If external management (inband or EXTM) is configured for the switch, they can ping each other via the external ports.
- LLDP does not see other switch modules that are connecting via the MM. That is, a chassis might have two or more switches that are installed in it and are all connected internally to the CMM, but they cannot see each other via LLDP through those MM connections.
- Although it might appear that a potential path might exist between the switch uplinks and the MM uplink, no such path exists. The embedded switches never send packets that come in via the switch uplinks over to the MM-facing interface, and vice versa. A loop cannot be introduced between the switch uplinks and the MM uplink.

Protected mode on embedded switches

Protected mode is an available option on embedded switches that is used to isolate switch management control from the MMs. Some customers are uncomfortable with the idea that the MM can reset a switch back to factory default settings or disable uplinks. You can enable the Protected mode feature to decouple the management of an embedded switch from the embedded MM.

For more information about enabling and configuring protected mode, see the documentation for the enclosure and switches. Consider the following points:

- Enabling Protected mode is a multistep process that involves enabling on the MM and switch module and then reloading the switch for it to take effect.
- Before the MM allows you to enable Protected mode, it requires that external management over uplinks for that switch module be enabled via the MM.

- Before you can enable Protected mode, you must configure an IP interface on the switch for use via the data ports or the EXTM port. You must connect via the console port or that external management interface when you attempt to enable Protected mode (to help ensure that you do not lock yourself out of switch management).
- When Protected mode is fully enabled, the MM cannot change the switch, and the MM uplink cannot be used as a path for switch management. The MM still can shut off the switch in the case of certain issues, such as an over temperature condition that is reported by the switch.
- Protected mode must be configured on a switch-by-switch basis; that is, it must be done for each switch that needs Protected mode.

Viewing MM, AMM, or CMM controlled features from the switch side

The team that manages the switch is often not the team that manages the MM. In many cases, the switch management team has no access to log in or manage the MM. Because the MM can control a number of aspects of the switch, it is important to be able to tell from the switch, how the MM is configured for these features. To receive valuable feedback from the switch about how the MM is configured for that switch, use the **`show system chassis`** command.

Example 7-11 shows the output of the **`show system chassis`** command as run on an EN4093 switch in a Flex System chassis (that uses a CMM in the enclosure).

Example 7-11 EN4093 switch, Flex System chassis, CMM, and show system chassis command

```
bay-1#show system chassis
Lenovo Flex System Chassis Related Information:

    Switch Module Bay         = 3
    Chassis Type              = Flex Enterprise
    Chassis UUID              = 4BED75FE9BF0381C86B64303066A3204
    POST Results              = 0xff

    Management Module Control -

        Default Configuration         = FALSE
        Skip Extended Memory Test     = FALSE
        Disable non-CMM Ports         = FALSE
        POST Diagnostics Control      = Normal Diagnostics

        Control Register              = 0x11
        Extended Control Register     = 0x00

    Management Module Status Reporting -

        Device PowerUp Complete       = TRUE
        Over Current Fault            = FALSE
        Fault LED                     = OFF
        Primary Temperature Warning   = OK
        Secondary Temperature Warning = OK

        Status Register               = 0x40
        Extended Status Register      = 0x01
```

BladeCenter based switches produce a similar output with this command. Consider the following points about the output that is shown in Example 7-11 on page 145:

- At the beginning of the output, the *Switch Module Bay* is shown, which indicates exactly in which bay the switch is installed. This information can help prevent confusion about which switch bay you are logged in to.
- Also at the beginning of the output is a Chassis UUID. This ID is a unique number for every enclosure and it can be used to confirm that you are logged in to a switch in the correct enclosure.
- The setting for *Disable non-CMM Ports* (called *Disable External Ports* in the BladeCenter) indicates whether the MM shut down the external ports for the switch. FALSE means that the MM did not shut down the ports. When the MM does shut down the EXT ports, you cannot enable the ports from the switch (and this output shows TRUE). If this output is TRUE, you must go into the MM and change the setting that controls it. In BladeCenter switches, external ports are disabled from the AMM by default and must be enabled from the AMM before you use the switch. With Flex System, the external ports are enabled in the default settings and optionally can be shut down from the CMM.
- The *Control Register* shows a hex value for a number of settings (in a cryptic fashion) with one important item being that the highest order bit of the second byte, which indicates whether the MM is locking out management over external ports (by default, management works only via the MM path). If you want to manage an embedded switch via any of the external uplinks or internal ports, you *must* change this setting in the MM. As shown in Example 7-11 on page 145, the *Control Register* is set to 0x11, which represents binary 0001 0001. For management over the uplinks to become operational, the high-order bit of the second byte (the “Y” in binary XXXX YXXX) must be showing as set to 1. For example, if this field is 0x19 (binary 0001 1001), management over the uplinks was enabled via the MM. Because the current value is 0x11 (binary 0001 0001), you know that the MM has this feature shut down for this device. Although you can see how the MM configured this feature from the switch, you *cannot change* this setting on the switch. It *must* be changed from the MM.

7.7.2 BladeCenter based switches

One of the unique differences between BladeCenter based switches and Flex System based switches is that the uplink ports on BladeCenter switches are disabled in the default settings via the MM or AMM. For a new installation, they must first be enabled once. After the port is enabled via the MM or AMM, control of enabling and disabling is then managed by the switch.

The Flex System switches also include an option to disable the uplinks of the switches via the CMM, but they are enabled in the default settings of the CMM.

7.7.3 Flex System based switches

Switch modules that are installed into a Flex System chassis also receive NTP and certain SNMP settings from the CMM or Flex System Manager (FSM). In the default settings, the NTP setting points back to the CMM and uses the CMM for time sync. This setting can be disabled by using the CMM and overriding the NTP control for the switch module. After this step, it is possible to go back into the switch and manually set the NTP. Until the CMM NTP control is released, changes to the NTP setting on the switch are overridden by the CMM after a brief period.

Some SNMP settings can also be pushed over via the FSM if one is present.

The default login option for the Flex System switches is to use secure protocols, such as SSH and HTTPS. Telnet and HTTP can be enabled, but are disabled in the default settings. This configuration is different from the BladeCenter embedded switches, which default to non-secure protocols, such as telnet and HTTP.

7.8 Deeper inspection of received control packets

When systems are first brought online, things often do not work out as expected. This section describes some useful commands for troubleshooting network-related problems and to identify some issues that are not related to the network.

7.8.1 Packet parsing (show mp packet)

The **show mp packet parse** command often is used to help troubleshoot packets that ingress and egress switch ports. A packet output can contain information, such as packet type (for example, TCP and UDP), time stamp, physical port number, VLAN ID, source and destination MAC, source and destination IP address, source and destination port number, and some header information. One of the most important technologies that can use this command is Easy Connect. Because packets are double tagged on ingress, this command can still see and present a window output of the inner tag information. Therefore, even if the configuration is using VLAN 4091 as the Easy Connect VLAN (for example), a **show mp packet parse rx port x** command can still see and provide window output information about packets that are tagged in other VLANs (for example, the inner VLAN). If packets with the EasyConnect VLAN 4091applied to a port are seen, it means that those packets are not being tagged by the source and are being seen on ingress with no tag on the frame.

Example 7-12 shows the output of a **show mp packet parse rx port x** command while the switch is in Easy Connect (Q-n-Q) mode. This output refers to a packet not being tagged as it is identified with VLAN 4091 (the outer tag VLAN in this example). If the VLAN ID indicates any other VLAN ID, it is being tagged from the source.

Example 7-12 Show mp packet parse command

```
354. Type: ARP Request, received 23:37:51 Thu Mar 15, 2001
     Port 19, VLAN 4091, Length 64, Reason 0x1000, Flags 0x0
     Dst MAC: ff:ff:ff:ff:ff:ff, Src MAC: 00:90:fa:5b:b5:08
     Sender MAC: 00:90:fa:5b:b5:08, Target MAC: 00:00:00:00:00:00
     Sender IP: 9.70.42.146 Target IP: 9.70.42.254
```

Example 7-12 is only one example of many ways that you can use a **show mp packet parse** command that is available to the switch. This command can be important for troubleshooting.

The following options can be used in a **show mp packet parse rx/tx** command:

- arp
- bgp
- bpdu
- cisco
- dhcp
- ecp
- fcoe
- ftp
- http
- icmp

- igmp
- ip-addr
- ipv4
- ipv6
- lacp
- lldp
- mac
- mgmtsock
- ntp
- ospf
- other
- pim
- port
- radius
- rarp
- raw
- rip
- snmp
- ssh
- tacacs
- tcp
- tcpother
- telnet
- tftp
- udp
- udpother
- vlan
- vrrp

7.8.2 Command data path consideration

Certain commands on Lenovo switches must have their path of execution defined (will it use the dedicated management port, a data port, or, for embedded switches, the MM uplink).

For example, the **ping** command, unlike for other switch vendors, uses the syntax **ping 10.10.10.10 data-port** to identify whether to use an interface on the data port or management port (or MM). The syntax that is used to ping a dedicated management port on a ToR switch (or the MM port on Flex System switches) is **ping 10.10.10.10 mgt-port** (it is also the default option if nothing is specified).

Example 7-13 shows the output of a **ping 10.10.10.10 ?** command on a top-of-rack switch (Flex System 4093 series switches have a third option, **extm-port**, for the external dedicated RJ45 port), which provides options to use with the **ping** command.

Example 7-13 Ping a data-port and an mgt-port

```
8264CS#ping 10.10.10.10 ?
  <0-4294967295>  Number of packets
  data-port       Data port
  mgt-port        Management port
  <cr>
```

Example 7-14 shows other useful ping options.

Example 7-14 Ping options

```
8264CS#ping
Enter hostname or IP address to ping: 10.10.10.10
Enter the port to use ["mgt"|"data"]:
Enter number of tries (0 for infinite) [5]:
Enter ping timeout value [1000]:
Enter payload size [0]:
Enter ping source address (if different from default own address):
Enter time to live [255]:
Enter type of service [0]:
Enter don't fragment bit (d/e) [disabled]:
```

Other commands also require defining what path to use, such as starting a telnet session from a RackSwitch or Flex System switch, or setting certain options, such as logging hosts for syslog support.

7.9 Port mirroring considerations

Port mirroring can be used to redirect packets from one port to another, usually for the purposes of capturing that data for analyses of some sort. Lenovo switches support a local form of this port mirroring. Consider the following points when you use this feature:

- A port mirror session consists of some number of source ports (called *mirroring-ports*), and a single destination port (called the *monitor-port*). Lenovo switches support 1 - 4 port mirror sessions, depending on the model type, version of code, and features enabled. For more information about how many sessions are available for a specific model, see the Application Guide for the version of code in use for that specific model.

  The number of mirroring (source) ports that are associated with a session is not restricted, with the exception that any physical port can belong to a single port-mirror session only, and a mirroring port cannot also be a monitoring port.

  The more mirroring-ports that are associated with a session, the greater the chance the monitor port becomes oversubscribed and of packets getting lost on the oversubscribed monitor port.

- Lenovo ports that are configured as monitor-ports still operate as normal ports. Therefore, it is possible to see flooded packets (broadcast, multicast, and unknown unicast) from other ports on the same VLAN as being allowed on the monitor port. It also participates in all normal port operations. This participation might affect the data that is being sent out the port (such as seeing some flooded packets twice: once from the mirror port and once from being flooded to the monitor port). To ensure that the monitor port sees only the mirrored data from the mirror-ports, ensure that the following perquisites are met:
  - Always place the monitor-port (port where the capturing device is attached) into an unused VLAN by setting the native/PVID/access VLAN to some unused VLAN on the switch.
  - To ensure that the capturing device is not affected by spanning-tree changes, set the monitor port to **`spanning-tree portfast`** (**`spanning-tree edge`** on older code).
- It is not necessary to enable tagging (**`switchport mode trunk`**) on the monitor port to pass tagged packets from the mirrored-ports out of the monitor port.

Leave the monitor (destination) port untagged to keep the configuration simple. It still passes all tagged and untagged traffic that is redirected from the mirror ports.

- Out-of-band management ports cannot be mirror or monitor ports. The port mirror feature works on data ports only.

For enclosure-based switches, the internal data ports or the external data ports can be used for port mirroring (as the monitor or mirror port).

7.10 LLDP recommendations

LLDP is enabled by default for current versions of Lenovo Networking products and can be enabled (by using the **`lldp enable`** command) on older versions of firmware that support LLDP. Other vendors almost universally support LLDP because it is a prerequisite for Fibre Channel over Ethernet (FCoE).

LLDP is sent on links between devices and at minimum, passes identifying information from each device to its immediately adjacent neighbor. Several vendors have similar proprietary protocols, which are not supported on Lenovo switches (for example, Cisco Discovery Protocol (CDP), Juniper (JDP), and Foundry (FDP), which is now part of Brocade).

The main use of LLDP is to ensure that the device at the other end of a link is the one that is expected to be there. This information is useful for troubleshooting and for initial deployment because it helps you verify that cables are placed where they are planned to be placed.

The most commonly used LLDP command is **`show lldp remote-device`**. (Other vendors, including Cisco, implement this function with a command syntax **`show lldp neighbors`** and newer Lenovo code also supports this format). Figure 7-9 shows an example of the display that is provided by this command.

```
sho lldp rem

LLDP Remote Devices Information
Legend(possible values in DMAC column) :
NB   - Nearest Bridge          - 01-80-C2-00-00-0E
NnTB - Nearest non-TPMR Bridge - 01-80-C2-00-00-03
NCB  - Nearest Customer Bridge - 01-80-C2-00-00-00
Total number of current entries: 5

LocalPort | Index | Remote Chassis ID          | Remote Port          | Remote System Name              | DMAC
----------|-------|----------------------------|----------------------|---------------------------------|---------
17        | 3     | 6c ae 8b bf fe 00          | 46                   |                                 | NB
18        | 2     | 6c ae 8b bf fe 00          | 45                   |                                 | NB
19        | 5     | 6c ae 8b bf 6d 00          | 44                   | TME1-Bay3                       | NB
20        | 4     | 6c ae 8b bf 6d 00          | 43                   | TME1-Bay3                       | NB
MGT       | 1     | fc cf 62 b3 49 00          | 11                   | TME1
```

Figure 7-9 Display from show lldp remote command

It is possible to drill down into entries in the LLDP display output by using the **show lldp remote-device <index-number>** command, in which the index number is shown in the second from left column of the output of the LLDP summary command. A sample of the output for a single link is shown in Figure 7-10.

```
G8264-Bottom#sho lldp rem 2
Local Port Alias: 18
   Remote Device Index     : 2
   Remote Device TTL       : 92
   Remote Device RxChanges : false
   Chassis Type            : Mac Address
   Chassis Id              : 6c-ae-8b-bf-fe-00
   Port Type               : Locally Assigned
   Port Id                 : 45
   Port Description        : EXT3

   System Name        :
   System Description : Lenovo Flex System Fabric EN4093 10Gb Scalable Switch(Upgrade2),
Lenovo Networking OS: version 7.7.5, Boot image: version 7.7.5
   System Capabilities Supported : bridge, router
   System Capabilities Enabled   : bridge, router

   Remote Management Address:
      Subtype             : IPv4
      Address             : 9.70.42.247
      Interface Subtype   : ifIndex
      Interface Number    : 128
      Object Identifier   :
```

Figure 7-10 Display from show lldp remote command for a specific device

7.10.1 Using LLDP

This section provides information about the use of LLDP.

Emulex NICs that run in VMware servers also send LLDP to the upstream switches, as shown in Figure 7-11.

```
bay-2#sh lldp rem 1
Local Port Alias: INTA1
        Remote Device Index     : 1
        Remote Device TTL       : 110
        Remote Device RxChanges : false
        Chassis Type            : Mac Address
        Chassis Id              : 00-90-fa-5b-af-15
        Port Type               : Mac Address
        Port Id                 : 00-90-fa-5b-af-15
        Port Description        :
        System Name        :
        System Description : Emulex OneConnect 10Gb Multi function Adapter
        System Capabilities Supported : station only
        System Capabilities Enabled   : station only
```

Figure 7-11 LLDP device display for a server with an Emulex adapter

More LLDP configuration commands are provided that add information to the LLDP protocol exchanges. These commands often are not necessary.

When two devices are connected with a static or LACP link aggregation, LLDP information is nonetheless sent on each individual link. It contains the identification of the port on the other device to which each link is connected.

For some Cisco devices, LLDP must be enabled by using the **`feature LLDP`** command.

One part of the output of the LLDP commands is the host name of the adjacent device. If there is no hostname configured on a Lenovo switch, this field is empty in the LLDP output. If possible, always configure the host name to ensure useful LLDP reporting.

7.11 Simple Network Management Protocol

SNMP is the most commonly used interface for managing devices by Network Management Platforms. SNMP provides well-defined methods that can be easily integrated into programs to monitor and configure the switches. The SNMP methods, which are referred to as objects, are defined in a Management Information Base (MIB), which is defined in a series of MIB files that are shipped with each firmware release.

7.11.1 Basic SNMP configuration items

There are a few infrastructure elements that are available for SNMP that provide switch information for the standard system objects. These elements also provide useful information that is used to uniquely identify the switch. It is a preferred practice to configure the objects that are shown in Table 7-5.

Table 7-5 Objects to configure

Object name	Description	Configuration command
sysName	Descriptive name to uniquely identify the switch.	**`snmp-server name`** (**`hostname`** also configures the **`snmp-server`** name and it is preferred to use **`hostname`**)
sysContact	Name and contact information for the entity that is responsible for managing this switch.	**`snmp-server contact`**
sysLocation	Location of the device. It is preferred to add data center and rack information, where applicable.	**`snmp-server location`**

7.11.2 SNMP v1/v2c

SNMP v1/v2c provides the simplest to implement SNMP services for the switch with minimal configuration. Access is limited to authentication by using Community strings and the data stream is not encrypted so it can be clearly decoded if there is access to the wire. Most corporate security policies limit SNMP v1/v2c access to read only.

SNMP v1 was the first implementation of SNMP as defined in the IETF RFCs 1155-1157. These RFCs defined the basic constructs of the SNMP MIBs and the procedures (get, getnext, set, and traps). SNMP v1 was quickly updated with SNMP v2 and v2c as defined by RFC 1905 and further refined with RFC 3416. This extension modified the MIB definition by increasing the counter sizes from 32 bits to 64 bits, which enabled improved device monitoring and defined bulk requests to optimize the procedures allowing for multiple requests to be included in a single request frame.

Enabling SNMP v1/v2c

SNMP v1/v2c access is disabled by default on the Flex System switches and enabled on the TOR switches. This process can be observed by running the **show snmp-server** command and looking for the Current v1/v2 access state. Example 7-15 shows the output of SNMP v1/v2c enabled.

Example 7-15 SNMP v1/v2C enabled

```
Current SNMP params:
  sysName:     "r11-EN4093-B"
  Read community string:  "ESSLabRead"
  Write community string: "ESSLabWrite"
  SNMP state machine timeout: 5 minutes
  Trap source address:  0.0.0.0
  SNMP Trap source loopback interface not set
  Authentication traps disabled.
  All link up/down traps enabled.

Current SNMP trap hosts:

Current v1/v2 access enabled

Current SNMPv3 USM user settings:
  1: name adminmd5, auth md5, privacy des
  2: name adminsha, auth sha, privacy des
  3: name v1v2only, auth none, privacy none
 17: name adminshaaes, auth sha, privacy aes

Current SNMPv3 vacmAccess settings:
  1: group name admingrp, model usm
     level authPriv,
     read view iso, write view iso, notify view iso
  2: group name v1v2grp, model snmpv1
     level noAuthNoPriv,
     read view iso, write view iso, notify view v1v2only

Current SNMPv3 vacmSecurityToGroup settings:
  1: model usm, user name adminmd5, group name admingrp
  2: model usm, user name adminsha, group name admingrp
  3: model snmpv1, user name v1v2only, group name v1v2grp
 17: model usm, user name adminshaaes, group name admingrp

Current SNMPv3 vacmViewTreeFamily settings:
  1: name v1v2only, subtree 1
     type included
  2: name v1v2only, subtree 1.3.6.1.6.3.15
     type excluded
  3: name v1v2only, subtree 1.3.6.1.6.3.16
```

```
     type excluded
  4: name v1v2only, subtree 1.3.6.1.6.3.18
     type excluded
  5: name iso, subtree 1
     type included

Current SNMP nlm settings:
  Nlm default log: enabled
  Global age out: 0
  Global entry limit: 0
```

To configure the SNMP access, use the following commands:

```
enable
configure terminal
snmp-server version v1v2v3
```

To disable the SNMP v1/v2c access, use the following commands:

```
enable
configure terminal
snmp-server version v3only
```

SNMP also can be enabled or restricted to read-only. By default, SNMP is enabled with read/write access. To configure the SNMP access, use the following commands:

```
enable
configure terminal
[no] access snmp [read-only | read-write]
```

To disable SNMP access, use the **`no access snmp`** the command.

For improved device security, it is recommended that SNMP access is restricted to SNMP v3 only. If SNMP v1/v2c access is required for your monitoring system, it is recommended to restrict SNMP access to read-only.

SNMP v1/v2c Security

SNMP v1/v2c read and write access is controlled by using community strings. These are four unique 32 character text strings with two each controlling read and write access, which are configured by using the following commands:

```
enable
configure terminal
snmp-server read-community STRING
[no] snmp-server read-community-additional STRING
snmp-server write-community STRING
[no] snmp-server write-community-additional STRING
```

The **`show snmp-server`** command displays the current setting of the community strings. By default, the read community strings are set to “public” and write community strings are set to “private”.

It is recommended to configure the read-community and write-community to unique values for the site or to disable the SNMP access.

Traps

Traps are used to alert a monitoring system, which provides server (receiver) services for SNMP traps (trap receiver) of error conditions. Most of the traps also are alerted with system log messages. SNMP v1/v2c traps are easy to configure by using the following command:

```
enable
configure terminal
snmp-server host IP_ADDRESS COMMUNITY_STRING
```

To configure the IP address and community string is the access string configured on the trap receiver. The trap server can be removed by using the following command:

```
no snmp-server host IP_ADDRESS
```

Link up and down traps are enabled by default and authentication traps alerting failed logins are disabled by default. It is recommended that authentication traps be enabled by using the following commands:

```
enable
configure terminal
snmp-server authentication-trap enable
```

It is also recommended to monitor only the link state of critical ports by using the following commands:

```
enable
configure terminal
[no] snmp-server link-trap PORT enable
```

Because port ranges are not allowed on these commands, an entry must be made for each port. Because traps on many ports can mask critical issues, it is recommended that link traps on non-critical ports are disabled.

The **`show snmp-server`** command can be used to display the current SNMP trap settings.

For more information about the supported SNMP traps for each firmware released, see the *Application Guide* for each release.

7.11.3 SNMP v3

SNMP v3 was developed to address security concerns that are not addressed in SNMP v1/v2c. The SNMP v3 protocol is defined in RFCs 3411-3418 and keeps the base SNMP v1/v2c protocol but extends it by adding encryption, integrity, and authentication protections.

SNMP v3 access control and packet encryption

SNMP v3 enables password authentication to determine whether the message is from a valid source that is based on configurable encryption for the password and encryption of the entire packet for privacy of the data. The authentication encryption can be configured as Hashed Message Authentication Code (HMAC) Message Digest Algorithm 5 (MD5) or Secure Hash Algorithm (SHA). The packet encryption is based on Data Encryption Standard (DES) with 56-bit encryption or Advanced Encryption Standard 128-bit (AES). Table 7-6 on page 156 lists the SNMP v3 security levels.

Table 7-6 SNMP v3 security levels

Level	Authentication	Encryption	Description
noAuthNoPriv	User name	None	Match the username for authentication only.
authNoPriv	Username and Password (MD5 or SHA)	None	Authentication with username and password that is based on MD5 or SHA algorithm.
authPriv	Username and Password (MD5 or SHA)	Yes (DES or AES)	Authentication with username and password that is based on MD5 or SHA algorithm and DES or AES packet encryption for privacy.

Enabling SNMP v3 access and security

If SNMP v3 is used, it is considered best practice to disable SNMP v1/v2c and to enable Boot Strict Mode to enable enhanced security and encryption. Use the commands that are shown in Example 7-16 for this configuration:

Example 7-16 Enabling Boot Strict Mode

```
WARNING:Enabling Boot Strict Mode will factory default the switch!  If you have
any configuration saved, you will need to save it to repost it.  The management IP
information will also be defaulted to make sure that you have a method to
reconnect to the switch.
enable
show running-config(copy and save to repost after booting)
configure terminal
boot strict enable
```

You are then prompted if you want to support old default users in strict mode. Answer **n** and only the default user `adminshaaes` is created. Next, reload the switch by using the **`reload`** command.

After the switch boots, paste any saved configuration back into the switch and disable SNMP v1/v2c by using the following command:

```
enable
configure terminal
```

Paste the previous configuration and then disable SNMP{ v1/v2c by using the **`snmp-server version v3only`** command.

Change the authentication password and encryption key for the default user `adminshaaes`, as shown in Example 7-17 on page 157.

Example 7-17 Changing the password and encryption key for the default user adminshaaes

```
snmp-server user 17 authentication-protocol sha authentication-password
Changing authentication password; validation required:
Enter current local admin password:<admin password>
Enter new authentication password (max 32 characters):<auth password>
Re-enter new authentication password:<auth password>
snmp-server user 17 privacy-protocol aes privacy-password
Changing privacy password; validation required:
Enter current local admin password:<admin password>
Enter new privacy password (max 32 characters):<privacy password>
Re-enter new privacy password:
```

If you selected to keep the old default users, you also must change the authentication and privacy passwords for user indexes 1 and 2.

Configuring SNMP v3 user access

Although configuring SNMP v3 access is more complex than configuring the SNMP v1/v2c community access, it does provide significantly enhanced security. The configuration can be simplified if it is broken down into the following steps:

1. The users can be displayed with the command show snmp-server v3 user, but this command does not show the user table index (1 - 17); therefore, it is recommended that the following command be used:

   ```
   show snmp-server | section user
   ```

   You must use lowercase **user** as the section in the command to yield the following output:

   ```
   Current SNMPv3 USM user settings:
     3: name v1v2only, auth none, privacy none
    17: name adminshaaes, auth sha, privacy aes
   ```

   If you enabled Boot Strict Mode, there is one user that is created by default. As described in “Enabling SNMP v3 access and security” on page 156, the authentication password and encryption key (privacy password) were changed for security.

   Create the user by entering the username, authentication type and password, and the privacy protocol and password in the SNMP v3 user table by using the following command:

   ```
   snmp-server user INDEX
   ```

   Where **INDEX** is an unused USM index with a value of 1 - 17 for the user table entry. The commands that are shown in Figure 7-12 on page 158 create a user that named `TestSNMPv3User1` as user index 4 and `TestSNMPv3User2` as user index 5 with authentication encryption of SHA and privacy encryption of AES-128.

```
snmp-server user 4 name TestSNMPv3User1
snmp-server user 4 authentication-protocol sha authentication-password
Changing authentication password; validation required:
Enter current local admin password:<admin password>
Enter new authentication password (max 32 characters):<authentication
password>
Re-enter new authentication password:<authentication password>
snmp-server user 4 privacy-protocol aes privacy-password
Changing privacy password; validation required:
Enter current local admin password:<admin password>
Enter new privacy password (max 32 characters):<privacy password>
Re-enter new privacy password:<privacy password>
snmp-server user 5 name TestSNMPv3User2
snmp-server user 5 authentication-protocol sha authentication-password
Changing authentication password; validation required:
Enter current local admin password:<admin password>
Enter new authentication password (max 32 characters):<authentication
password>
Re-enter new authentication password:<authentication password>
snmp-server user 5 privacy-protocol aes privacy-password
Changing privacy password; validation required:
Enter current local admin password:<admin password>
Enter new privacy password (max 32 characters):<privacy password>
Re-enter new privacy password:<privacy password>
```

Figure 7-12 Commands to create TestSNMPv3User1 and TestSNMPv3User2

The user must be created before it can be added to a group; therefore, this step must be completed first.

2. Assigns a user to a group name by using an entry in the VacmSecurityToGroup table. The group name is used to map access privileges in the VacmAccess table. The current groups can be displayed by using the **show snmp-server v3 group** command; however, this command does not show the group table index (1 - 17); therefore, it is recommended that the following command is used:

   `show snmp-server | section Group`

   You must use `Group` as the section in this command to yield the following output:

   ```
   Current SNMPv3 vacmSecurityToGroup settings:
     3: model snmpv1, user name v1v2only, group name v1v2grp
    17: model usm, user name adminshaaes, group name admingrp
   ```

   There are two default user names assigned group names. To create a group to define a user, use the following command:

   `snmp-server group` *`INDEX`*

   Where INDEX is an unused value 1 - 17 for the group table entry. Although the group index does not have to be the same index as the user index, keeping them the same reduces the complexity of the configuration. You assign the `user-name` and `group-name`, which maps an entry in the access privilege table. The `group-name` can be used for multiple user names. The following commands create a user group that is named `SNMPv3User` to map the new users:

   ```
   snmp-server group 4 group-name SNMPv3User
   snmp-server group 4 user-name TestSNMPv3User1
   snmp-server group 4 security usm
   ```

```
snmp-server group 5 group-name SNMPv3User
snmp-server group 5 user-name TestSNMPv3User2
snmp-server group 5 security usm
```

3. Views define MIB trees that are assigned for access in the access table. Each view is defined by a view name and can consist of multiple entries (indexes) in the vacmViewTreeFamily table to include or exclude subtrees. There is a default view named iso that includes the entire MIB.

   The views can be displayed by using the **show snmp-server v3 view** command; however, this command does not show the view table index (1 - 128)' therefore, it is recommended that the following command is used:

   ```
   show snmp-server | section View
   ```

   You must use `View` as the section in the command to yield the following output:

   ```
   Current SNMPv3 vacmViewTreeFamily settings:
     1: name v1v2only, subtree 1
        type included
     2: name v1v2only, subtree 1.3.6.1.6.3.15
        type excluded
     3: name v1v2only, subtree 1.3.6.1.6.3.16
        type excluded
     4: name v1v2only, subtree 1.3.6.1.6.3.18
        type excluded
     5: name iso, subtree 1
        type included
   ```

   There are two default views. To create a view, you must create entries in unused indexes and use a previously undefined view name. The following commands create a view that is named `TestView`, including the base ObjectID (OID), 1, and excludes the system table:

   ```
   snmp-server view 6 name TestView
   snmp-server view 6 tree 1
   snmp-server view 6 type included
   snmp-server view 7 name TestView
   snmp-server view 7 tree 1.3.6.1.2.1.1
   snmp-server view 7 type excluded
   ```

4. Access defines what SNMP objects a group of users can access and the type of security the users follow. The access name is the same name as the group name for which the access is defined and the SNMP object access is defined by using access names from the view table. Access is defined for read, write, and notify (traps) objects.

   The access entries can be displayed by using the **show snmp-server v3 access** command; however, this command does not show the access table index (1 - 128); therefore, it is recommended that the following command is used:

   ```
   show snmp-server | section Access
   ```

   You must use `Access` as the section in the command to yield the following output:

   ```
   Current SNMPv3 vacmAccess settings:
     1: group name admingrp, model usm
        level authPriv,
        read view iso, write view iso, notify view iso
     2: group name v1v2grp, model snmpv1
        level noAuthNoPriv,
        read view iso, write view iso, notify view v1v2only
   ```

There are two default access entries. The following commands create an access entry for the new users (`TestSNMPv3User1` and `TestSNMPv3User2`) that are defined in the group SNMPv3User and provide user access to the view TestView:

```
snmp-server access 3 security usm
snmp-server access 3 level authPriv
snmp-server access 3 name SNMPv3User
snmp-server access 3 read-view TestView
snmp-server access 3 write-view TestView
snmp-server access 3 notify-view TestView
```

If these steps are completed, two SNMP v3 users are created with the security level *authPriv*, which means that the user must authenticate with a password that is encrypted by using SHA and the packet is encrypted for privacy by using AES 128-bit. These users have access to the entire MIB, except for the system subtree.

SNMP v1 trap configuration

SNMP v1 traps can be configured by using the **`snmp-server host`** command, as described in section 7.11.2, “SNMP v1/v2c” and 7.11.3, “SNMP v3”, or by creating entries in the SNMP v3 configuration tables. To create the entry in the SNMP v3 tables, entries must be created in the user (USM), group (VacmSecurityToGroup), view (vacmViewTreeFamily), access (VacmAccess), notify, target-address, target-parameters, and community tables.

Complete the following steps to create the configuration:

1. Create a username to be used in the trap with no authentication and no privacy by using the following commands:

   ```
   snmp-server user 6 name TestV1Trap
   snmp-server user 6 authentication-protocol none
   snmp-server user 6 privacy-protocol none
   ```

2. Add the user to a group by using the following commands:

   ```
   snmp-server group 6 name V1Trap
   snmp-server group 6 user-name TestV1Trap
   snmp-server group 6 security snmpv1
   ```

3. Create a view, if needed. For simplicity, it is recommended to use the default view iso.

4. Assign the access that links the group name to a view by using the following commands:

   ```
   snmp-server access 6 name V1Trap
   snmp-server access 6 security snmpv1
   snmp-server access 6 notify-view iso
   ```

5. Create an entry in the notify table to select the management targets to receive the trap alerts. The entry is a table index. Use the same index as the corresponding group entry for simplicity. Because the entry also contains a name that is a unique identifier for the notify table entry, it is recommended that the same name is used at the group name. There is a tag to identify an entry in the target address table (again use the group name for simplicity) by using the following commands:

   ```
   snmp-server notify 6 name V1Trap
   snmp-server notify 6 tag V1Trap
   ```

   Use the **`show snmp-server | section notify`** command to display the notify table.

6. Create an entry in the target address table to identify the IP address of the trap receiver. For simplicity, use the same index value as used in the notify table. This entry has a name that is the same value as tag in the notify table and an IP address of the trap receiver. The entry also contains a parameter-name that identifies an entry in the target-parameters table, as shown in the following commands:

   ```
   snmp-server target-address 6 name V1Trap address 172.16.6.101
   snmp-server target-address 6 parameters-name V1Param
   snmp-server target-address 6 taglist V1Param
   ```

   Use the **show snmp-server | section Addr** command to display the target-address table. More trap receivers can be defined with the same parameters by creating more target-address entries that use a different index, name, and IP address.

7. Create an entry in the target-parameters table with the name the same as the parameters-name in the target-address table. For simplicity, use the same index as used in the target-address table. Configure the user name to point to the USR table entry (from step 1) and select the security as snmpv1, as shown in the following commands:

   ```
   snmp-server target-parameters 6 name V1Param
   snmp-server target-parameters 6 user-name TestV1Trap
   snmp-server target-parameters 6 security snmpv1
   ```

   Use the **show snmp-server | section Param** command to display the target-parameters table.

8. Create an entry in the community table to define the SNMP community string that is transmitted with the trap. Use the same index as used in the target-address table. Configure a value for index to uniquely identify the community entry. It is recommended to use the same name as the target-address entry or the USR table user name. Configure the name as the community string that is passed to the trap receiver for authentication. Configure the user-name to point to the USR table entry (from step 1), as shown in the following commands:

   ```
   snmp-server community 6 index V1Trap
   snmp-server community 6 name YourCommunityString
   snmp-server community 6 user-name TestV1Trap
   ```

   Use the **show snmp-server | section community** command to display the community table.

SNMP v2c trap configuration

SNMP v2c traps can be configured just like SNMP v1 traps, except changing security to snmpv2.

Complete the following steps to create the configuration:

1. Create a username to be used in the trap with no authentication and no privacy by using the following commands:

   ```
   snmp-server user 7 name TestV2Trap
   snmp-server user 7 authentication-protocol none
   snmp-server user 7 privacy-protocol none
   ```

2. Add the user to a group by using the following commands:

   ```
   snmp-server group 7 name V2Trap
   snmp-server group 7 user-name TestV2Trap
   snmp-server group 7 security snmpv2
   ```

3. Create a view, if needed. For simplicity, it is recommended to use the default view iso.

4. Assign the access that links the group name to a view by using the following commands:

```
snmp-server access 7 name V2Trap
snmp-server access 7 security snmpv2
snmp-server access 7 notify-view iso
```

5. Create an entry in the notify table to select the management targets to receive the trap alerts by using the following commands:

```
snmp-server notify 7 name V2Trap
snmp-server notify 7 tag V2Trap
```

6. Create an entry in the target address table to identify the IP address of the trap receiver by using the following commands:

```
snmp-server target-address 7 name V2Trap address 172.16.6.101
snmp-server target-address 7 parameters-name V2Param
snmp-server target-address 7 taglist V2Param
```

7. Create an entry in the target-parameters table by using the following commands:

```
snmp-server target-parameters 7 name V2Param
snmp-server target-parameters 7 user-name TestV2Trap
snmp-server target-parameters 7 security snmpv2
```

8. Create an entry in the community table to define the SNMP community string by using the following commands:

```
snmp-server community 7 index V2Trap
snmp-server community 7 name YourCommunityString
snmp-server community 7 user-name TestV2Trap
```

SNMP v3 trap configuration

SNMP v3 traps can be configured as are SNMP v1 and V2c traps, except changing security to usm, configuring the user (USR) table entry with the appropriate authentication and privacy settings, and configuring appropriate level on the access table entry.

Complete the following steps to create the configuration:

1. Create a username to be used in the trap with no authentication and no privacy, as shown in Figure 7-13.

```
snmp-server user 8 name TestV3Trap
snmp-server user 8 authentication-protocol sha authentication-password
Changing authentication password; validation required:
Enter current local admin password:<admin password>
Enter new authentication password (max 32 characters):<authentication
password>
Re-enter new authentication password:<authentication password>
snmp-server user 8 privacy-protocol aes privacy-password
Changing privacy password; validation required:
Enter current local admin password:<admin password>
Enter new privacy password (max 32 characters):<privacy password>
Re-enter new privacy password:<privacy password>
```

Figure 7-13 Creating a username to be used in the trap with no authentication and no privacy

2. Add the user to a group by using the following commands:

```
snmp-server group 8 name V3Trap
snmp-server group 8 user-name TestV3Trap
snmp-server group 8 security usm
```

3. Create a view, if needed. For simplicity, it is recommended to use the default view iso.
4. Assign the access that links the group name to a view and setting the proper level by using the following commands:

```
snmp-server access 8 name V3Trap
snmp-server access 8 level authPriv
snmp-server access 8 security usm
snmp-server access 8 notify-view iso
```

5. Create an entry in the notify table to select the management targets to receive the trap alerts by using the following commands:

```
snmp-server notify 8 name V3Trap
snmp-server notify 8 tag V3Trap
```

6. Create an entry in the target address table to identify the IP address of the trap receiver by using the following commands:

```
snmp-server target-address 8 name V3Trap address 172.16.6.101
snmp-server target-address 8 parameters-name V3Param
snmp-server target-address 8 taglist V3Param
```

7. Create an entry in the target-parameters table with the appropriate security level for the user by using the following commands:

```
snmp-server target-parameters 8 name V3Param
snmp-server target-parameters 8 user-name TestV3Trap
snmp-server target-parameters 8 security usm
snmp-server target-parameters 8 level authPriv
```

7.11.4 Other security considerations

It is recommended that issues that are described in Chapter 6, “Securing access to the switch” on page 115 be considered. Management networks (MNet/MMask) and Management ACLs must be used on switches that are attached to public networks to further isolate the switch from threats.

It is also important to remember to use the **`access snmp read-only`** command to restrict access to monitor only the switch if SNMP is not used to modify the configuration.

7.11.5 Troubleshooting the SNMP configuration

When troubleshooting SNMP, the first thing that you must do is verify that the SNMP requests are actually reaching the switch. The easiest way to verify this information is to review the SNMP counters that are retrieved by using the **show snmp-server counters** command, as shown in Figure 7-14.

```
------------------------------------------------------------------
SNMP statistics:
snmpInPkts:              20685   snmpInBadVersions:              0
snmpInBadC'tyNames:         18   snmpInBadC'tyUses:              0
snmpInASNParseErrs:          0   snmpEnableAuthTraps:            2
snmpOutPkts:             20661   snmpInBadTypes:                 0
snmpInTooBigs:               0   snmpInNoSuchNames:              0
snmpInBadValues:             0   snmpInReadOnlys:                0
snmpInGenErrs:               0   snmpInTotalReqVars:            82
snmpInTotalSetVars:          0   snmpInGetRequests:              0
snmpInGetNexts:             82   snmpInSetRequests:              0
snmpInGetResponses:          0   snmpInTraps:                    0
snmpOutTooBigs:              0   snmpOutNoSuchNames:             0
snmpOutBadValues:            0   snmpOutReadOnlys:               0
snmpOutGenErrs:              0   snmpOutGetRequests:             0
snmpOutGetNexts:             0   snmpOutSetRequests:             0
snmpOutGetResponses:       105   snmpOutTraps:                   0
snmpSilentDrops:             0   snmpProxyDrops:                 0
```

Figure 7-14 Example output of the show snmp-server counters command

The `snmpInPkts` counter is key to determining whether SNMP requests are received by the switch. You also must closely review `snmpInBadC'tyNames` to determine whether the requester is using the correct community string for an SNMP v1/v2c request.

If SNMP v3 is used and the `snmpInPkts` and `snmpOutPkts` counters are incrementing without a counter for the SNMP procedure (Get, Set, and so on) and an error is being received by the client, verify that the authentication and privacy encryption passwords are used correctly.

Another useful tool is to use the **show mp packet parse rx snmp** command to determine the addresses of the SNMP clients that are sending requests.

7.12 Quality of service

This Lenovo Press book does not provide information about CoS, ToS, DiffServ, and DSCP. Instead, it provides a basic description of how to use quality of service (QoS) for Lenovo Networking products. For more information about specific features, see the technical documentation for your switch at the following website:

http://ibm.com/support/entry/portal/Documentation

QoS is the ability for a network switch with QoS functions to set values and prioritize packets to ensure traffic based on a priority value over the priority of other traffic types. QoS prioritization is applied to ports with the use of ACLs. The Lenovo Networking products can apply only CoS values on ingress. The Lenovo switches also support ranges; however, to add a range, it must be converted to hex. For example, if the requirement of a certain QoS value to prioritize a range of ports is **`permit udp any any range 2326 2373`**, the equivalent value in a Lenovo that is running revision 7.*x* is **`access-control list [x] tcp-udp destination-port 2326 0x2f`** (x re to the ACL number assigned).

The default DSCP values that are set within the Lenovo switches are listed at the end of this section. These typical default values can also be seen in other vender switches. The default values can be changed to whatever is required for use.

Example 7-18 shows an output of the default **`qos dscp`** value settings.

Example 7-18 Output of a show qos dscp value command

```
G8264-1#show qos dscp
Current DSCP Remarking Configuration: On

  DSCP    New DSCP  New 802.1p Prio
--------  --------  ---------------
    0         0          0
    1         1          0
    2         2          0
    3         3          0
    4         4          0
    5         5          0
    6         6          0
    7         7          0
    8         8          1
    9         9          0
   10        10          1
   11        11          0
   12        12          1
   13        13          0
   14        14          1
   15        15          0
   16        16          2
   17        17          0
   18        18          2
   19        19          0
   20        20          2
   21        21          0
   22        22          2
   23        23          0
   24        24          3
   25        25          0
   26        26          3
   27        27          0
   28        28          3
   29        29          0
   30        30          3
   31        31          0
   32        32          4
   33        33          0
   34        34          4
   35        35          0
   36        36          4
   37        37          0
```

```
38          38          4
39          39          0
40          40          5
41          41          0
42          42          0
43          43          0
44          44          0
45          45          0
46          46          5
47          47          0
48          48          6
49          49          0
50          50          0
51          51          0
52          52          0
53          53          0
54          54          0
55          55          0
56          56          7
57          57          0
58          58          0
59          59          0
60          60          0
61          61          0
62          62          0
63          63          0
```

Although Example 7-18 on page 165 can be overwhelming, it is possible that a small subset of these values might ever be used in a production environment. The following subset examples are available:

- DSCP value of 0 with an 802.1p Priority of 0 = high value data
- DSCP value of 10 with an 802.1p Priority of 1 = standard application data
- DSCP value of 18 with an 802.1p Priority of 2 - enhanced application data
- DSCP value of 26 with an 802.1p Priority of 3 = session initiation protocol (sip data)

Up to 8 802.1p Priority values can be configured for use.

7.12.1 Configuring QoS with examples

In this section, some examples of how an RackSwitch G8264 switch can prioritize certain types of data traffic are described.

In this first example (which is taken from the *subset example list* in 7.12, “Quality of service” on page 164), the DSCP value 0 is used to prioritize high volume data. This traffic might be considered to be the highest required type of data to be prioritized in an enterprise environment.

Example 7-19 shows a sample configuration of a QoS configuration to prioritize high value data and it is applied on what are referred to as *dirty ports*. Dirty ports can be assumed to be facing servers and or inter-switch link ports that might span between a pair of vLAG switches.

Example 7-19 An example of a QoS configuration to prioritize high-volume data traffic

```
G8264-1 Configuration:
access-control list 31 tcp-udp destination-port 1500
access-control list 31 re-mark in-profile dscp 0
access-control list 31 action permit
```

```
!
access-control list 32 tcp-udp source-port 1500
access-control list 32 re-mark in-profile dscp 0
access-control list 32 action permit
!
access-control list 33 tcp-udp destination-port 1501
access-control list 33 re-mark in-profile dscp 0
access-control list 33 action permit
1
access-control list 35 tcp-udp source-port 1501
access-control list 35 re-mark in-profile dscp 0
access-control list 35 action permit
!
access-control group 1 list 31
access-control group 1 list 32
access-control group 1 list 33
access-control group 1 list 34
!
interface port 1-8,17-24 (dirty ports)
        dscp-remark
        access-control group 1
qos dscp re-marking (globally enables qos dscp re-marking)
```

In this next example (which also is taken from the *subset example list* in 7.12, “Quality of service” on page 164), the DSCP value 3 is used to prioritize SIP traffic for VoIP services.

Example 7-20 shows an example of prioritizing SIP traffic to ensure a higher prioritization, which ensures the highest quality of voice services.

Example 7-20 An example of a QoS configuration to prioritize VoIP SIP traffic

```
access-control list 40 tcp-udp destination-port 5060
access-control list 40 re-mark in-profile dscp 26
access-control list 40 action permit
!
access-control list 41 tcp-udp destination-port 5061
access-control list 41 re-mark in-profile dscp 26
access-control list 41 action permit
!
access-control group 10 list 41
access-control group 10 list 42
!
interface port 1-8,17-24 (dirty ports)
        dscp-remark
        access-control group 10
!
qos dscp re-marking (globally enables qos dscp re-marking)
```

In Example 7-19 on page 166 and Example 7-20, ports 1 - 8 are used as the inter-switch link (ISL) and ports 17 - 24 are used as ports that directly face the servers.

Although the ACLs have an implicit allow statement at the end, you are not building a list of ACLs to allow or deny traffic. Instead, you are prioritizing traffic.

7.12.2 Access Control List

Access Control List (ACL) can be used to help provide security through the switch. It is not intended to be used to secure management to and from the switch via SNMP, SSH, and so on. For more information about securing a switch management access, see 7.12.3, “Management Access Control List” on page 169.

When you set up ACLs to allow or deny traffic, there is an implicit allow statement at the end so a deny all must be configured at the end of the ACL list to drop all other access. For more information about options, see the technical documentation for your switch at this website:

http://ibm.com/support/entry/portal/Documentation

This section provides only a basic description of how ACLs can be used. ACLs can use several different options to provide for securing access to resources on one side or the other. Options can be used, such as source and destination MAC, source and destination IP (and protocol type), and source and destination TCP/UDP port. Much like QoS ACLs, ACLs can be applied on ingress only.

Example 7-21 shows a configuration example of an allowed access list for securing SSH from a specific host based on IP and TCP port number only followed by a deny all statement at the end.

Example 7-21 Example of an access list to SSH only from a single host

```
access-control list 1 ipv4 source-ip-address 192.168.20.10 255.255.255.255
access-control list 1 tcp-udp destination-port 22 0xffff
access-control list 1 action permit
!
access-control list 10 tcp-udp destination-port 22 0xffff
access-control list 10 action deny
!
interface port 19
        access-control list 1
        access-control list 10
!
```

Although Example 7-21 creates only a simple access list, a more complete and restrictive list can be defined. If you plan to create several dozen ACLs, you can use ACL groups to simplify the configuration. Instead of applying dozens of ACLs to a number of ports, you can create a simple group with the numbers of ACLs to be added, and that single group can be applied to the required ports.

Example 7-22 shows how adding a list of ACLs to a single group and applying that group to a number of ports can minimize the configuration.

Example 7-22 Example of an access group being used to simplify the configuration

```
access-control list 1 ipv4 source-ip-address 192.168.20.10 255.255.255.255
access-control list 1 tcp-udp destination-port 22 0xffff
access-control list 1 action permit
!
access-control list 10 tcp-udp destination-port 22 0xffff
access-control list 10 action deny
!
access-control group 1 list 1
access-control group 1 list 10
```

```
!
interface port 19
        access-control group 1
!
```

7.12.3 Management Access Control List

Management ACLs (MACLs) are used to secure the local management protocols, such as Telnet, SSH, SNMP, and so on. MACLs are similar to that of normal ACLs, except that MACLs are automatically applied to the out-of-band management ports on switches that support it (that is, G8264 interface port mgt). Up to 256 MACLs can be defined by using the **access-control macl <macl number> ?** command. For more information about options, see the *Application Guide* at the following website and search for “macl”:

http://ibm.com/support/entry/portal/Documentation

However, if the requirement to secure the switch management is solely based on source IP only, the use of the management network might be a better solution. MNET and MMASK can be used to secure access to the switch processor by defining access with subnets or even specific end hosts. For more information about options and limitations, see the *Command Line Interface Guide* at the following website and search for “management network”:

http://ibm.com/support/entry/portal/Documentation

7.13 Network Time control considerations

Establishing a common time source for devices in a network is important for several reasons. One of the more important reasons is log file message synchronization; that is, when an entry has a time stamp in the log, it can be compared to events in other devices that occurred at the same time.

Lenovo switches offer a local clock, but it is not battery-backed and returns to the default setting if a reload occurs. These switches also support an NTP client, and in some models, Precision Time Protocol (PTP). As with other features, always check the application guide that is associated with your switch model and code version for more information about what features are available.

To ensure consistent log time stamps, it is recommended to configure at least one NTP server and a backup NTP server (if one is available). Lenovo switches support a maximum of two NTP servers.

For more information about commands and options for setting NTP options on Lenovo switches, see the appropriate CLI reference for the model of switch and version of code in use. At a minimum, this process includes the following tasks:

- Configuring a primary NTP server
- Enabling NTP

Although this configuration is not part of NTP, if you want to represent a local time instead of a universal time, you must also set a time zone on the switch by using the **system timezone** command.

The following common commands are available for reviewing NTP:

- **`show ntp`**: A snapshot of how NTP is configured.
- **`show ntp counters`**: Shows counts for requests, responses, and updates for NTP servers.
- **`show ntp associations`**: Shows any servers with which the switch is associated.
- **`show clock`**: Shows current time on the switch (as affected by NTP and any configured Daylight Saving Time and time zone set).

7.14 sFlow considerations and issues

sFlow is an industry standard protocol that is supported on Lenovo switching products. It functions by inspecting a configurable subset of the packets and gathers statistics on them. These statistics are periodically reported to a collector whose address is configured in the switch. The following commands are available for configuring sFlow (for more information, see the Command Reference manuals for the various products):

- **`sflow server <ip address>`** sets the address of the collector server. This address can be an IPv4 address only, not IPv6. The default setting is to reach the server via the management port. If the collector server is to be reached by the data port, the command must be followed by the **`data-port`** operand (and **`extm-port`** for Flex System switches with a dedicated EXTM port in use for this purpose).
- **`sflow enable`** enables sFlow processing on the switch.
- **`sflow polling`** sets the polling interval (5 - 60 seconds) for data on each specific port, which must be configured on each port where data is to be collected.
- **`sflow sampling`** sets the packet sampling rate, which is the average of how many packets flow in to the port for each packet that is chosen for sampling. This rate is 1 in 256 packets to 1 in 65536 packets.

The potential issue with sFlow is that it runs as a process on the management processor chip on the switches. It is necessary to be careful not to put excessive load on the Management Processor (MP); therefore, the frequency of sampling and reporting by using sFlow must be limited. sFlow is not supported in stacking mode because the MP on the master switch in a stack must manage the entire stack.

Consider the following preferred practice limits on the sFlow parameters:

- sFlow sampling is limited to no more frequently than one packet in every 256. Even this setting can generate significant CPU loading because the first 128 bytes of each packet are forwarded to the collector server. The sampling frequency must be limited by the number of ports that are being monitored by sFlow and the level of traffic on those ports. Typically, this limit is no more frequent than one packet in 1000.
- sFlow polling can be configured to occur at intervals of from once in 5 seconds to once in 60 seconds. Less frequent polling is less taxing on the management processor; however, it might be prudent to set the switches to be monitored less frequently to avoid overloading the collector depending on the number of switches that is being monitored by the collector.

7.15 Understanding Control Plane Policing

All Flex System embedded switches and RackSwitch top-of-rack switches support control plane policing (CoPP). CoPP keeps the CPU of a switch from being overwhelmed by certain traffic types (for example, too many ARP packets destined for the processor), which can leave the processor too busy to process more critical packets and lead to unstable operation of the switch.

The CoPP QoS values are best left at the default values, but there can be times when it might be necessary to tune some of these values. Normally, the only time to consider tuning the CoPP values is when messages are received that indicate the limit is being reached for one of the queues for no obvious or known reason. For example, you might receive log messages that indicate *Protocol control discards*, as shown in Example 7-23.

Example 7-23 "Protocol control discards" message

```
Aug 19  3:50:11 Switch1 ALERT   system: Protocol control discards: arp-bcast or
ipv6-nd  packets are received at rate higher than 200pps,hence are discarded on
queue 5!
```

Even if these types of messages are received, look for underlying causes that indicate that a networking issue is being encountered before you tune these values. Example 7-23 is the result of a device that is putting out 50,000 ARPs per second in the upstream network. In that case, the CoPP was successfully protecting the CPU from being overwhelmed with processing that amount of ARPs. Avoid tuning the values higher in cases where they are doing their job protecting the processor from such a network issue. Instead, fix the network issue that is causing the CoPP to engage.

When CoPP is actively discarding, it is affecting *only* traffic that is headed *to* the processor, not *through* the switch. As shown in Example 7-23, although the processor was protected, the ARPs still flooded to all ports that were carrying the VLAN that the ARPs were on. The hosts that are attached to these ports mostly crashed while they attempted to process such a high rate of ARP packets. However, because of CoPP, the switch remained stable and administrable to make it possible to troubleshoot this portion of the network. (In this case, troubleshooting is not possible without CoPP.) CoPP protects the switch processor. To protect the hosts in our example, apply Storm Control on the uplinks in the direction of the storm to throttle the incoming ARPs to a level that the hosts can handle.

If CoPP must be configured, the following command provides a syntax example:

```
qos protocol-packet-control rate-limit-packet-queue 5 400
```

This example tunes the queue (from the previous example) from taking effect at 200 pps to not taking effect until it reaches 400 pps of this type of traffic.

Important: Tune the CoPP values *only* when you understand *exactly* why CoPP is being triggered. Setting values too high can lead to processor saturation and unstable switch operation. There is a reason that these values are set to the default settings and you must use caution when you change these settings.

For more information about configuring and controlling CoPP, see the *Application Guide* for your switch model and code version.

7.16 Verifying an implementation

It is a preferred practice to ensure that proper installation and operation is verified, even after the initial physical and configuration of an installation is completed. Do not assume that because a ping can pass through the switch that all is operational.

Verification of any implementation can start with Layer 1 physical connectivity. Next, check the configuration and operation of Layer 2 VLANs and Spanning Tree operations and other specific configuration requirements that are needed to support the installation and operation of a newly added element.

7.16.1 What to look at first

Things to look for during installation, or any additions to a network, can include (but not limited to) the following items:

- Proper cable placement. Make sure that the cables are probably seated and connected to the intended ports.
- Interface link status. Ensure that the port comes up to ensure that both ends have matching characteristics. It is also best to ensure that the ports are not shut down while you test connectivity.
- Proper functioning transceiver. Incorrect transceivers on either end typically cause no link status or might also cause a port to be shut down on one end.
- Correct interface speed, duplex, and flow control. Ensure correct port configuration on both ends of the wire to ensure correct function and the wanted performance.
- VLAN configuration and VLAN port assignments. Ensure that VLANs are enabled and assigned to respective interfaces for port-to-port communication is typically one of the biggest issues that is seen during initial installations.
- Native VLAN ID assignment. Matching native VLAN assignment ensures proper communication, which results in communication of non-tagged frames between two switch ports on ports that have tagging and trunking enabled.
- Spanning Tree configuration. Ensure that Spanning Tree is properly configured to interoperate within the environment to increase stability and provide proper failover conditions. An improper Spanning Tree configuration can cause network disruption or even a complete network failure.
- Correctly configured and working vLAG. vLAG is quickly becoming one of the most deployed active/active solutions and is required for most installations. Proper installation and validation of vLAG can have significant performance benefits in any environment.
- Correct LACP and 802.1AX-2008 (formerly IEEE 802.3ad) proper configuration. Ensure that LACP is properly configured and running to help prevent network disruptions that are caused by broadcast storms (also referred to as network loops).

7.16.2 Common commands to verify operational status

Although a **`show run`** output can provide a quick summary of the configuration of a switch, it does not provide the information that is required to ensure operational status of a new installation.

The commands that are described in this section (with the example output of each) can be used to ensure proper operational status. They can provide information to help determine potential improper configuration steps. When you verify the issues that are listed in 7.16.1, “What to look at first” on page 172, you can use the commands that are described in this section to verify proper operational status.

LLDP

You can use the **show lldp remote port** command to help determine proper physical link connectivity between remote peers.

Example 7-24 shows the results of a **show lldp remote port** command, which can be useful for new or added environmental switch components. Always run this command soon after you start a new port to ensure that proper cabling is completed.

Example 7-24 Example show lldp remote port output

```
G8264-1#show lldp remote port
LLDP Remote Devices Information
Legend(possible values in DMAC column) :
NB   - Nearest Bridge          - 01-80-C2-00-00-0E
NnTB - Nearest non-TPMR Bridge - 01-80-C2-00-00-03
NCB  - Nearest Customer Bridge - 01-80-C2-00-00-00
Total number of current entries: 10

LocalPort | Index | Remote Chassis ID         | Remote Port          | Remote System Name               | DMAC
----------|-------|---------------------------|----------------------|----------------------------------|---------
1         | 2     | 74 99 75 41 3b 00         | 1                    | G8264-2                          | NB
1         | 3     | 74 99 75 41 3b 00         | 1                    |                                  | NnTB
5         | 4     | 74 99 75 41 3b 00         | 5                    | G8264-2                          | NB
5         | 5     | 74 99 75 41 3b 00         | 5                    |                                  | NnTB
17        | 6     | 00 00 c9 e9 7b 45         | 00-00-c9-e9-7b-45    |                                  | NB
18        | 8     | 00 00 c9 e9 7a 99         | 00-00-c9-e9-7a-99    |                                  | NB
29        | 9     | 00 25 03 19 d3 00         | 49                   | G8000                            | NB
63        | 10    | 00 18 b9 7c 85 80         | Gi0/1                | CoreSwitch                       | NB
MGT       | 1     | 74 99 75 41 3b 00         | 65                   | G8264-2                          | NB
MGT       | 7     | 00 25 03 19 d3 00         | 44                   | G8000                            | NB
```

Example 7-25 shows a **show lldp remote port [*x*]** command (*x* is the physical port number) that can provide more information, such as a full description of the remote device, its version of code, and its management IP address

Example 7-25 Example show lldp remote port 63 output

```
G8264-1#show lldp remote port 63
Local Port Alias: 63
        Remote Device Index     : 10
        Remote Device TTL       : 112
        Remote Device RxChanges : false
        Chassis Type            : Mac Address
        Chassis Id              : 00-18-b9-7c-85-80
        Port Type               : Interface Name
        Port Id                 : Gi0/1
        Port Description        : GigabitEthernet0/1

        System Name        : CoreSwitch
        System Description : Cisco IOS Software, C3560 Software
(C3560-IPSERVICES-M), Version 12.2(53)SE2, RELEASE SOFTWARE (fc3)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2010 by Cisco Systems, Inc.
Compiled Wed 21-Apr-10 05:33 by prod_rel_team
        System Capabilities Supported : bridge, router
        System Capabilities Enabled   : bridge, router
```

```
        Remote Management Address:
                Subtype             : IPv4
                Address             : 192.168.0.104
                Interface Subtype   : system port number
                Interface Number    : 1
                Object Identifier   :
                Object Identifier   :
```

Interface link status

The **show interface status** command can display the results of link conditions.

Example 7-26 shows the results of the link status on several ports. It includes information such as speed, duplex, flow control, link condition, and description.

Example 7-26 Results of a show interface status command

```
G8264-1(config)#show interface status
------------------------------------------------------------------------
Alias   Port   Speed    Duplex     Flow Ctrl      Link     Description
------- ----   -----   --------  --TX-----RX--   ------   -------------
1         1     40000     full      no     no       up        ISL/Peer-Link
5         5     40000     full      no     no       up        ISL/Peer-Link
9         9     40000     full      no     no      down       9
13        13    40000     full      no     no      down       13
17        17    10000     full      no     no       up        ESXi111
18        18    10000     full      no     no       up        ESXi112
19        19     1000     full      no     no    disabled     dhcp-client
20        20     1000     full      no     no       up        dhcp-server
```

Properly functioning transceiver

The **show transceiver** command provides less information in more recent code revisions. However, it typically displays information, such as link, transceiver type, vendor, part number, and acceptance. In newer code revisions, the **show interface port [x] transceiver details** command must be run to show detailed information.

Example 7-27 shows the results of a **show transceiver** command. It is a preferred practice to run this command during initial configuration and when you are troubleshooting a physical link problem.

Example 7-27 Results of a show transceiver command

```
G8264-1#show transceiver

Port             Link  Transceiver    Vendor           Part             Approve
---------------- ----- -------------- ---------------- ---------------- --------
1 QSFP+ 1        LINK  UnEqQD 1.0m    BLADE NETWORK    BN-QS-QS-CBL-1M  Accepted
5 QSFP+ 2        LINK  UnEqQD 1.0m    BLADE NETWORK    BN-QS-QS-CBL-1M  Accepted
9 QSFP+ 3        < NO Device Installed >
13 QSFP+ 4       < NO Device Installed >
17 SFP+  1       LINK  PasDAC 1.0m    IBM-Amphenol     90Y9425-N28500A  Approved
18 SFP+  2       LINK  PasDAC 1.0m    IBM-Amphenol     90Y9425-N28500A  Approved
19 SFP+  3       Down  Cu_SFP         Blade Network    BN-CKM-S-T       Approved
20 SFP+  4       LINK  Cu_SFP         Blade Network    BN-CKM-S-T       Approved
```

Example 7-28 shows the results of the **show interface port [x] transceiver details** command. This command can provide more information, such as light levels, part number, date, and serial number.

Example 7-28 Results of a show interface port 63 transceiver detail command

```
Port             TX  Link  TXFlt Volts DegsC TXuW  RXuW  Transceiver    Approve
---------------- --- ----- ----- ----- ----- ----- ----- -------------- --------
63 SFP+ 47       Ena LINK  -N/A- -N/A- -N/A- -N/A- -N/A- Cu_SFP         Approved
         Blade Network    Part:BN-CKM-S-T       Date:080710 S/N:BNT082808X
```

VLAN configuration and VLAN port assignments

In older revisions of VLAN code, configuration issues might occur during initial configuration or when more VLANs are added post production. If these issues are overlooked, you can spend hours troubleshooting a simple mistake. For example, if a newly created VLAN is added to an interface, the switch does not enable the VLAN. In newer revisions of code, this scenario is no longer an issue. The switch enables the VLAN and assigns it to a new spanning tree group if PVRST+ is used.

Example 7-29 shows an example of the **show vlan** command. You can use it to display the VLAN name, status, and current port members.

Example 7-29 Results of a show vlan output command

```
G8264-1(config)#show vlan
VLAN    Name                             Status     Ports
----  -------------------------------  ------  -------------------------
1     Default VLAN                     ena     9 13 21-28 30-62
10    MGMT-NET                         ena     1 5 17-19 29
20    vMotion                          ena     1 5 17 18 20 29
30    DATA-NET                         ena     1 5 17 18 29
999   Native                           ena     29
4090  ISL-Trunk                        ena     1 5
4093  VLAN 4093 (INTERNAL)             ena     64
4094  VLAN 4094 (INTERNAL)             ena     63
4095  Mgmt VLAN                        ena     MGT
```

Example 7-30 shows an example of a **show vlan [*x*] information** command, which can be used to display more information about that VLAN.

Example 7-30 Results of a show vlan 10 information command

```
G8264-1(config)#sh vlan 10 information
Current VLAN 10:
    Name "MGMT-NET", ports 1 5 17-19 29, enabled,
    Vports : empty
    Protocol: empty,
    Spanning Tree 10
    Current VLAN VMAP Config is empty
    Vlan creation time: 0 days, 0 hours, 1 minute and 29 seconds.

    Flooding settings:
    Flood of unregistered IPMC: ena
    Send unregistered IPMC to CPU: ena
    Optimized flooding: dis
```

Native VLAN ID assignment

The Native VLAN ID is used when a switch port is configured as a Trunk port (Multiple VLANs) with a requirement of a single, non-tagged VLAN. For example, in a VMware ESXi environment all traffic (including VM Data and vMotion) often features tagging enabled on the vSwitch. However, if you do not want to tag the ESXi Host Management interface, the connected switch port can be set to use a native VLAN.

Example 7-31 shows the native VLAN configuration of a switch port that requires a single VLAN to not be tagged.

Example 7-31 Results of a show interface trunk 17 command

```
G8264-1#show interface trunk 17
Alias    Port Tag    Type    RMON Lrn Fld Openflow PVID     DESCRIPTION  VLAN(s)
              Trk                                 NVLAN
-------- ---- --- ---------- ---- --- --- -------- ------ -------------- -----------------------
17       17    y  External    d    e   e      d      10   ESXi111        10 20 30

* = PVID/Native-VLAN is tagged.
# = PVID is ingress tagged.
Trk  = Trunk mode
NVLAN = Native-VLAN
```

Spanning Tree configuration

In the default settings, all Lenovo switches have Spanning Tree enabled (if supported) and in PVRST+ mode. PVRST+ mode interoperates with Cisco Rapid PVST+ mode to help ensure rapid convergence and per VLAN STP control.

Spanning Tree has several important features that must be understood to ensure a proper integration into a Spanning Tree environment. The task is to ensure that the bridge priorities on all leaf switches (anything other than the Spanning Tree core switches) are set to 32768 (default) or raised to its highest level of 65536 (highest), while any device you want to become the root has its priority set lower than the 32K default.

Example 7-32 shows the output of a **show run | section spanning-tree** command to identify the configured bridge priority of a Spanning Tree group (if no value is set, it is using the default 32K).

Example 7-32 Output of a show run | section spanning-tree

```
G8264-1#show run | section spanning-tree
spanning-tree stp 1 vlan 1
!
spanning-tree stp 10 bridge priority 4096
spanning-tree stp 10 vlan 10
!
spanning-tree stp 20 bridge priority 4096
spanning-tree stp 20 vlan 20
!
no spanning-tree stp 26 enable
spanning-tree stp 26 vlan 4090
!
spanning-tree stp 30 bridge priority 4096
spanning-tree stp 30 vlan 30
!
spanning-tree stp 110 bridge priority 4096
spanning-tree stp 110 vlan 999
```

Example 7-33 shows the output of a **show spanning-tree bridge** command to identify all VLAN Spanning Tree assignments, whether they are using the default settings.

Example 7-33 Output of a show spanning-tree bridge command

```
G8264-1#show spanning-tree bridge

STG       Priority    Hello     MaxAge    FwdDel    Protocol          VLANs
------    --------    ------    ------    ------    --------    ------------------
1         32768       2         20        15        PVRST       1 4093 4094
10        4096        2         20        15        PVRST       10
20        4096        2         20        15        PVRST       20
26        32768       2         20        15        PVRST       4090
30        4096        2         20        15        PVRST       30
110       4096        2         20        15        PVRST       999
128       32768       2         20        15        PVRST       4095
```

Example 7-34 shows the output of a **show spanning-tree blockedports** command, which can help quickly identify all blocked ports on a switch.

Example 7-34 Output of a show spanning-tree blockedports command

```
G8264-1#show spanning-tree blockedports

Instance   Blocked Port List
---------- ---------------------
1          43

Number of blocked ports (segments) in the system :1
```

vLAG is configured and functioning properly

vLAG is used to provide for cross-switch aggregations (splitting a single aggregation between a pair of switches). vLAG uses a Virtual MAC Address that is shared as a single common MAC to identify to a remote system. This configuration enables the use of a single aggregation (also known as PortChannel or EtherChannel) to be created without requiring the use of Spanning Tree to block redundant connections.

Note: All vLAG paired switches within a common Layer 2 environment *must* contain a unique Tier ID from other vLAGed paired switches. For more information about vLAG,4.1, “Virtual Link Aggregation Group considerations” on page 48.

Example 7-35 shows the output of a **show vlag information** command. This command can be used to display potential issues across a pair of switches that are running vLAG during a new installation for a newly added vLAG device.

Example 7-35 Output of a show vlag information command

```
G8264-1(config)#show vlag information
vLAG Tier ID: 10
vLAG system MAC: 08:17:f4:c3:dd:09
Local MAC 74:99:75:41:23:00 Priority 0 Admin Role PRIMARY (Operational Role PRIMARY)
Peer MAC 74:99:75:41:3b:00 Priority 0
Health local 192.168.0.91 peer 192.168.0.92 State UP
ISL trunk ID 65
ISL state    Up
Auto Recovery Interval:  300s (Finished)
```

```
Startup Delay Interval:  120s (Finished)

vLAG 65: config with admin key 1029, associated trunk 66, state formed
```

Proper LACP and 802.1AX-2008 (formerly IEEE 802.3ad) configuration

LACP is commonly used between switches and between switch and server to provide a dynamic aggregation between the devices, to increase the available the bandwidth beyond that of a single link, and offer greater high availability (a single link failure does not cause a loss of connection).

Example 7-36 shows the results of a **show lacp information** command. This command can help you identify issues or problems with negotiation between two devices.

Example 7-36 Output of a show lacp information command

```
G8264-1(config)#show lacp information
port    mode    adminkey  operkey   selected   prio  aggr  trunk  status  minlinks
--------------------------------------------------------------------------------
1       active       105      105    yes      32768     1     65    up        1
5       active       105      105    yes      32768     1     65    up        1
9       off            9        9     no      32768    --     --    --        1
13      off           13       13     no      32768    --     --    --        1
17      off           17       17     no      32768    --     --    --        1
18      off           18       18     no      32768    --     --    --        1
19      off           19       19     no      32768    --     --    --        1
20      off           20       20     no      32768    --     --    --        1
```

Example 7-37 shows the results of a **show port [x] lacp counters** command. This command can help to determine whether the switch is sending LACPDUs but is not receiving them from the remote device (or vice versa). Typical cases where LACP is not negotiating properly are when the remote end is not configured to use LACP or when one of the two ends is set to **short** (1 second) and the other is set to **long** (30 seconds) timers. A conflict in short and long settings can cause LACP instability between peers.

Example 7-37 Output of a show interface port 29 lacp counters command

```
G8264-1(config)#sh interface port 29 lacp counters
------------------------------------------------------------------
LACP statistics for port 29:
Valid LACPDUs received         : 593
Valid Marker PDUs received     : 0
Valid Marker Rsp PDUs received : 0
Unknown version/TLV type       : 0
Illegal subtype received       : 0
LACPDUs transmitted            : 591
Marker PDUs transmitted        : 0
Marker Rsp PDUs transmitted    : 0
```

7.17 Lenovo support process

This section describes how to open a support ticket on Lenovo Networking switches.

7.17.1 Information gathering

Problems that occur in a network often are difficult to troubleshoot and solve. Providing the following information to Support personnel helps to resolve the situation more quickly:

- A full and accurate description of the problem. This information is the most important piece of information to give Support. The following details should be provided:
  - The frequency of the failure.
  - Which nodes in the network fail and which ones do not.
  - Any error messages that are seen on individual nodes.
- The topology of the relevant section of the network.
- The Show Tech output for all switches that are involved in the problem.

It is also helpful to find a representative server and provide the following information:

- Operating system version
- NICs that are installed
- NIC firmware and driver versions installed
- Any non-default NIC configuration (for example, teaming configuration and vNICs)

Upload these support files to this website:

`http://www.ecurep.ibm.com/app/upload`

7.17.2 Opening a ticket

Use one of the following methods to open a support ticket:

- For all Lenovo and IBM branded networking devices via phone, call 1-800-IBM-SERV
- For all Lenovo or IBM top-of-rack switches, see the following IBM Service Request website:

  `https://www-947.ibm.com/support/servicerequest/HwHome.action`

7.17.3 Entitlement information required to open a support ticket

For the support methods that are listed in 7.17.2, “Opening a ticket”, the model type and serial number of the switch is needed.

If the Switch is accessible via SSH or Telnet, run the **`show version`** command. As shown in Figure 7-15 on page 180, the Machine type (MTM) and Electronic Serial Number (ESN) are displayed above the software version information.

```
RS G8264#sh version
System Information at 13:25:02 Wed Oct  9, 2001
Time zone: No timezone configured
Daylight Savings Time Status: Disabled

IBM Networking Operating System RackSwitch G8264, Stack

Switch has been up for 0 days, 0 hours, 44 minutes and 30 seconds.
Last boot: 12:41:22 Wed Oct  9, 2001 (reset from Telnet/SSH)

MAC address: 74:99:75:41:23:00    IP (If 1) address: 0.0.0.0
Management Port MAC Address: 74:99:75:41:23:fe
Management Port IP Address (if 128): 192.168.0.91
Hardware Revision: 0
Hardware Part No:  BAC-00065-00
Switch Serial No:  Y010CM29Y0NU
Manufacturing date: 12/40

MTM Value: 7309-HC4
ESN: 23B7573
Software Version 7.6.1.0                    (FLASH image2), factory default configuration.
```

Figure 7-15 MTM and ESN information by running show version command

The information often also can be found on a label on the bottom of the switch, as shown in Figure 7-16.

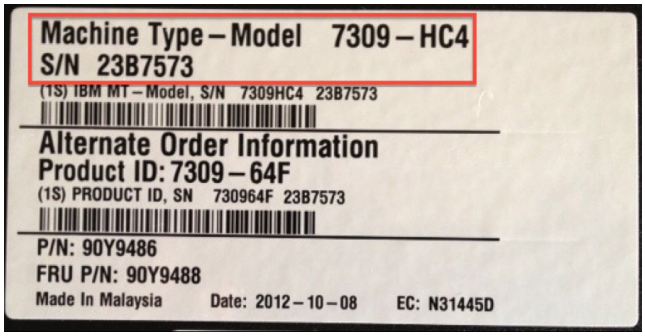


Figure 7-16 MTM and ESN label on bottom of switch

7.18 Command-line parsing with the pipe option

Searching the output of CLI commands is critical for rapidly reviewing vast amounts of returned information. This section describes how to search command output by using *command piping*. This feature is available with the following products (which must be configured to run isCLI mode, if they are not already):

- All G7*XXX* to-of-rack switches (all releases)
- Flex System embedded switch models EN4093, EN4093R, EN2092, CN4093, and SI4093 (all releases)
- All G8*XXX* top-of-rack switches (Lenovo, IBM, or BNT branded), and any BladeCenter embedded switches that are labeled as BNT, IBM, or Nortel, that are running 7.*X* or later code (for some of these products, some later 6.*X* releases also have this feature)

Although all of the top-of-rack switches run isCLI in the default settings, the default settings for embedded switches in the BladeCenter and previous Flex System solutions use the menu-driven CLI that does not support this piping feature.

For switches that are running the menu CLI, this setting can be changed to run isCLI by logging in to the menu CLI by running the **/boot/prompt ena** command, logging out, and then logging back in. When you log back in, you are prompted to select a CLI. Select isCLI to obtain access to the isCLI and its associated piping functions. This change to isCLI mode can be made permanent by running the s **boot cli-mode iscli** and **no boot cli-mode enable** isCLI commands.

Tip: If you want to receive the prompted login option, ensure that no other user is logged in. If the switch is set for prompted mode and a user is logged in to the menu or isCLI modes, any other user who attempts to log in is automatically placed into the current user's CLI mode, even if CLI prompted mode is enabled.

7.18.1 Basic searching

The fundamental element of command piping is the pipe symbol "|", which is the vertical bar symbol on the keyboard that separates the isCLI command and one of the following options to use grep-like functions on output:

- **include**: Search output for matching lines
- **exclude**: Search output for non-matching lines
- **section**: Search output for matching section
- **begin**: Show output beginning at first match

For example, use the command that is shown in Example 7-38 to search the output of the **show run** command for all strings that include the word "interface".

Example 7-38 Using the pipe option - Basic usage

```
show run | include interface
```

The following fundamental rules for syntax apply:

- Unless you are otherwise instructed (see 7.18.2, "Advanced searching" on page 182), all search strings that you enter are case-sensitive.
- If the wanted string includes spaces, the string that is searched must include surrounding double quotation marks (for example, **show run | include "interface port"**).
- Some special characters (that is, the asterisk or the slash) must be handled with special consideration (see 7.18.2, "Advanced searching" on page 182).

Of the various pipe options, only the **section** option does not have an obvious function. The **section** option is similar to the **include** option, except that instead of displaying *all* matching lines, it matches *only* lines that are left-aligned in the output. It also includes subcommands that are related to the left-aligned matching lines. For example, the **show run | section ip** command output might resemble the output that is shown in Example 7-39.

Example 7-39 Use of the pipe section option

```
Switch#show run | section ip
portchannel hash source-ip-address
!
interface ip 1
        ip address 10.1.1.1 255.255.255.0
        enable
        exit
!
```

Two matching left-aligned lines (**portchannel hash source-ip-address** and **interface ip 1**) are returned. However, because the **interface ip 1** command also includes associated subcommands (the IP address, and so on), it shows the related lines from that **section**.

7.18.2 Advanced searching

Beyond simple string searches, case sensitivity can be removed in the search. OR and AND searches can be performed, in addition to multiple OR and AND searches on the same line. You must use regular expression syntax for much of this function. For more information, see this website:

http://tmml.sourceforge.net/doc/tcl/re_syntax.html

Example of controlling case sensitivity of the search

Enclosing the uppercase and lowercase version of a character in left and right brackets [ ] tells the parser to look for either case of that character. Example 7-40 shows how to look for the string “auth”, the string “Auth”, and either of these strings by removing the case sensitivity for the first letter “a”.

Example 7-40 Case sensitive searching

```
Switch#show system | inc auth
RADIUS authentication currently OFF
TACACS+ authentication currently OFF

Switch#show system | inc Auth
LDAP Authentication currently OFF
  Authentication traps disabled.

Switch#show system | inc [Aa]uth
RADIUS authentication currently OFF
TACACS+ authentication currently OFF
LDAP Authentication currently OFF
  Authentication traps disabled.
```

Example of an OR search

OR functions can be done by adding more pipe signs between strings to search the whole string must be wrapped in double quotation marks if there are any spaces in the search).

An OR search syntax is shown in the following example:

```
show run | inc string1|string2
```

Example 7-41 on page 183 shows an OR syntax in use. It returns lines that contain the string “auth” or the string “access”.

Example 7-41 OR searching on a string

```
Switch#show system | inc auth|access
RADIUS authentication currently OFF
TACACS+ authentication currently OFF
HTTP access currently enabled on TCP port 80
HTTPS server access currently enabled on TCP port 443
NETCONF access currently enabled
SNMP access currently read-write
Telnet/SSH access configuration from BBI currently disabled
Telnet access currently enabled on TCP port 23
```

As shown in Example 7-42, one of the strings in the OR features spaces; therefore, it includes the quotation marks that surround the entire set of strings.

Example 7-42 Or searching on strings with spaces

```
Switch#show system | inc "auth|access conf"
RADIUS authentication currently OFF
TACACS+ authentication currently OFF
Telnet/SSH access configuration from BBI currently disabled
```

The OR function can be strung together with multiple pipes and strings, as shown in Example 7-43.

Example 7-43 Or searching on multiple strings

```
Switch#show system | inc TACACS|RADIUS|Telnet|LDAP
RADIUS authentication currently OFF
TACACS+ authentication currently OFF
LDAP Authentication currently OFF
Telnet/SSH access configuration from BBI currently disabled
Telnet access currently enabled on TCP port 23
```

Note: The entire search must be wrapped n double quotation marks if there are any spaces in any of the strings.

Example of an AND search

Performing an AND function is also part of regular expression searches. You can use an AND function with syntax, such as **`string1.*string2`**. The **`.*`** (period followed by an asterisk) says to search for two strings that are separated by at least one character on a line (for more information about how this syntax works, see 7.18.2, “Advanced searching” on page 182). An AND search syntax is shown in the following example:

```
show run | inc string1.*string2
```

Example 7-44 on page 184 shows searching with the OR function first, and then the use of the AND function instead to reduce the output. First, searching with the OR function to look for the string `Port` OR the string `disabled`.

Example 7-44 OR search

```
Switch#show system | inc Port|disabled
MGMT-A Port MAC Address: fc:cf:62:12:21:fe
MGMT-A Port IP Address (if 127):
MGMT-B Port MAC Address: fc:cf:62:12:21:ef
MGMT-B Port IP Address (if 128):
Use of BOOTP for configuration currently disabled
ErrDisable recovery disabled, timeout 300 sec
Port-based Port Mirroring currently disabled
  Authentication traps disabled.
  oper     - disabled   - offline
  strong password status: disabled
User configuration from BBI currently disabled
Telnet/SSH access configuration from BBI currently disabled
```

Example 7-45 shows how to replace the OR pipe in the same command with what is an AND function (`.*`). This substitution finds only lines that have the string `Port` *and* the string `disabled`, in that order.

Example 7-45 Replacing the same OR search with an AND search

```
Switch#show system | inc Port.*disabled
Port-based Port Mirroring currently disabled
```

The AND syntax that is shown in Example 7-45 matches *only* if the string `Port` comes *before* the string `disabled`. It is possible to search for either order of the two strings by including both orders ORed on the same line, as shown in Example 7-46. (In this case, the output is the same as Example 7-45. However, if there is a match in the reverse order, it also is displayed.)

Example 7-46 Replacing the same OR search with an AND search

```
Switch#show system | inc Port.*disabled|disabled.*Port
Port-based Port Mirroring currently disabled
```

Double quotation marks must be placed around the entire search if there are any spaces. Example 7-46 also shows the advanced function of having ORs and ANDs on the same line.

More advanced searches

This section describes more advance searches. Some of this information is based on hypothetical searches; however, the concepts can be applied to do some advanced searches.

Example 7-47 and Example 7-48 on page 185 shows how to filter the output on a specific CoPP queue (queue 19) for the counter display command.

Example 7-47 shows the full output.

Example 7-47 Raw output for QoS queue command

```
RS G8052#show qos protocol-packet-control queue-counters
|--------------|-----------------|
| Packet Queue | Received Packets|
|--------------|-----------------|
| 0            | 119011          |
|--------------|-----------------|
| 5            | 1685718         |
|--------------|-----------------|
```

```
| 8            | 933860          |
|--------------|-----------------|
| 11           | 538             |
|--------------|-----------------|
| 16           | 1838592         |
|--------------|-----------------|
| 19           | 1320            |
|--------------|-----------------|
| 22           | 0               |
|--------------|-----------------|
| 23           | 0               |
|--------------|-----------------|
| 24           | 0               |
|--------------|-----------------|
| 25           | 0               |
|--------------|-----------------|
| 29           | 36001           |
|--------------|-----------------|
| 31           | 4775            |
```

Example 7-48 shows output that is filtered to show only queue 19.

Example 7-48 Searching for queue 19 only (the command line is wrapped)

```
RS G8052#show qos protocol-packet-control information queue | include "[|] 19 +[|]
[0-9a-z]+ +"
| 19           | 1200       |
```

This regular expression searches for the specific pipe character by including it between square brackets, followed by a space, followed by the queue number (19), then a *space* and a **+** sign, which means one or more spaces must follow after the queue number, another pipe, another space, then the group **[0-9a-z]+**, which means any alphanumeric char one or multiple times (might resume to numerics only because the counters are numeric, but this was chosen for example purposes). The last part of the regular expression searches for multiple spaces after the alphanumeric value (this example does not include the last pipe character because a regular expression does not have to fully match a line unless explicitly specified with **^** and *$* delimiter in line-by-line matching).

Note: To enter a literal question mark anywhere in isCLI, you must press Ctrl+v, then enter the ?. Regex uses the question mark often, but it can be avoided with (match_group){0,1} if you want to paste regular expressions from a text document.

Lenovo switches do not support metasyntax (such as modifiers to specify that the regular expression search is performed without sensitivity to case). For partial case insensitive searches, you can use artifices, such as **[Cc]at catches [Mm]ouse**.

For example, for This AND That matching, if you start with a string that you want to search for, such as the following string:

```
That some garbled 123 this text is tricky
```

If you want a regex expression that matches both **this** *and* **that** in the string example (in any order of occurrence), the search resembles the following example:

```
(aTt]oteoperahis.*[Tt]hat)|([Tt]hat.*[Tt]his)
```

If any of the two words are missing in the searched text, the regular expression no longer matches.

The Regex Buddy tool can be helpful for regular expression syntax. For more information, see this website:

http://www.regexbuddy.com/

8

Converged networking

Many customers use converged networks to improve return on investment for infrastructure. Converged infrastructures can present some unique challenges to deploying these solutions. This chapter provides some insight about ways to meet these challenges that might not be included in common equipment documentation.

This chapter focuses on storage convergence and includes the following topics:

8.1 FCoE Considerations

This chapter describes preferred practices when Fibre Channel over Ethernet (FCoE) is used in the network.

8.1.1 General considerations

As described in 7.5, “Firmware upgrade considerations” on page 137, switch firmware must be flashed with FCoE actively configured to ensure that the Fibre Channel ASIC onboard the 8264CS and CN4093 switches is properly updated.

Ensure that the server that is sending FCoE data has the current firmware and drivers installed on the Converged Network Adapter (CNA). Adapter vendors made significant changes in FCoE implementation over the past few years. Running older FCoE drivers and firmware is a significant cause of unexpected behavior and poor performance.

8.1.2 FCoE and FCFs

The FCoE protocol allows servers to use a single network connection to enable access to TCP/IP and storage resources. Access to storage is accomplished by encapsulating the Fibre Channel protocol inside an Ethernet frame.

When hosts send FCoE traffic to a switch, the primary task for the switch is to move the FCoE packets to the nearest FCF. When the packet arrives at an FCF, the Fibre Channel protocol frame is de-encapsulated and sent to a Fibre Switch. Switches, such as the EN4093, G8124, and G8264, provide FCoE functionality only. The CN4093 and G8264CS provide FCoE and can function as an FCF.

8.1.3 NPV versus full fabric modes

When a switch is functioning as the FCF, the FCF must be configured to operate in one of two modes: NPV or Full Fabric.

If storage is connected directly to an 8264CS or CN4093, the switch must be configured for its FCF to operate in Full Fabric mode. Zoning and all other Fibre Channel administrative tasks must be performed on the 8264CS or CN4093.

When the 8264CS or CN4093 switch is connecting to a Fibre Channel Fabric, the FCF can be configured in NPV or Full fabric mode. NPV mode is the preferred configuration because it allows all Fibre Channel administration to be performed on the Fibre Channel switches.

8.1.4 Zoning

No issues or special considerations exist relative to zoning and Lenovo networking. Hard or soft zoning can be implemented on the Fibre Channel switch.

When zones are created, the preferred configuration is to have a single initiator and single target per zone. When a device is added or removed from a fabric, the fibre switch sends RSCN events to all devices in the zone. Active hosts in the zone that are not entering or leaving can have latency effects when this situation occurs. Creating a zone of one initiator and one target prevents this situation from occurring.

8.1.5 Limitations and scaling

Converging Ethernet and Fibre Channel networking requires implementing a Fibre Channel engine inside the Ethernet switch. As with Ethernet, too many nodes on the same FCoE VLAN can diminish performance. When a network solution is implemented in which an 8264CS or CN4093 is functioning as the FCF, care must be taken to avoid overloading a single VLAN with FCoE traffic. To maximize total system performance and avoid problems that are associated with a congested Fibre network (latency, fibre login failures, and so on), create FCoE VLANs as the number of physical FCoE hosts increases.

Example 8-1 shows the guidelines for an 8264CS or CN4093 operating in NPV or full fabric mode.

Example 8-1 Guidelines for an 8264CS or CN4093 operating in either NPV or full fabric mode

```
< 30 hosts:  1 VLAN with up to 12 uplinks to the Fabric
31-70 hosts:  2 VLANs with up to 6 uplinks to the Fabric
71-140 hosts:  4 VLANs with up to 3 uplinks to the Fabric
141-160 hosts:  6 VLANs with up to 2 uplinks to the Fabric
```

The number in Example 8-1 corresponds to *physical* hosts. The number of *virtual* hosts that are running on each physical host does not affect the guidelines.

Figure 8-1 shows multiple FCoE VLANs.

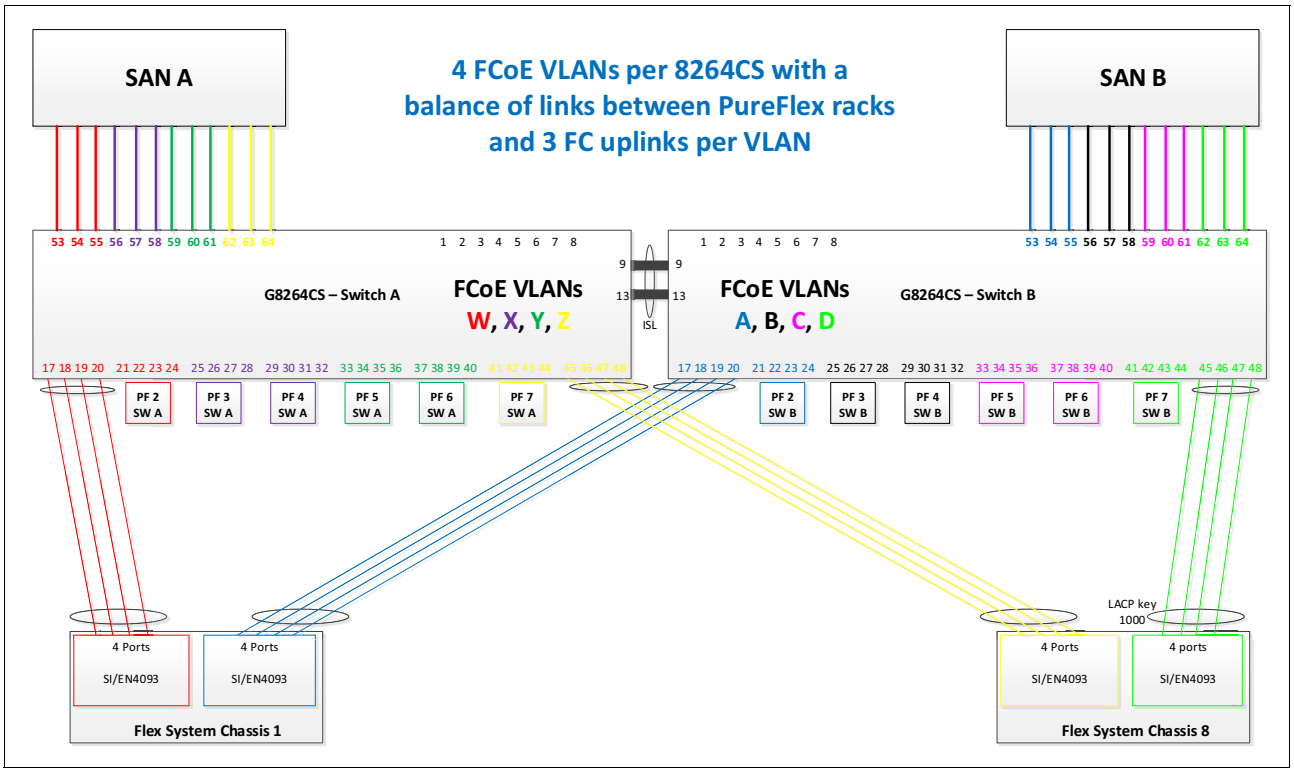


Figure 8-1 Multiple FCoE VLANs

8.2 Ethernet SAN considerations

Considerations the following points regarding any Ethernet-based SAN:

- Reliability: Data paths must be redundant and handle failures efficiently.
- Performance: A high-speed and uncongested network must support Jumbo Frames.
- Latency: Cut through performance and high-speed links is important.

The Lenovo Networking 10 Gbps and 40 Gbps Ethernet switches satisfy all of these requirements by using the redundancy, link aggregation, and jumbo Ethernet packet features. Virtual Link Aggregation (VLAG) is another feature that improves reliability and performance.

Another performance-enhancing feature to consider is Converged Enhanced Ethernet (CEE) if every device, end-to-end, supports CEE, including the NICs and Switches. CEE improves the Ethernet traffic with early congestion notification and traffic metering features to even out the flow of any traffic that is using high bandwidth. This feature can be useful in eliminating slow TCP traffic patterns that result from traffic loss because of congestion and retransmissions.

A TCP retransmission results in a slow start congestion avoidance that is defined in the TCP protocol. Repeated retransmissions because of congestion results in what is known as a *saw tooth traffic pattern*, as shown in Figure 8-2. The use of CEE can reduce the maximum relative throughput, but eliminates the saw tooth pattern. Enabling CEE on Lenovo Networking switches is accomplished by using the **`cee enable`** command.

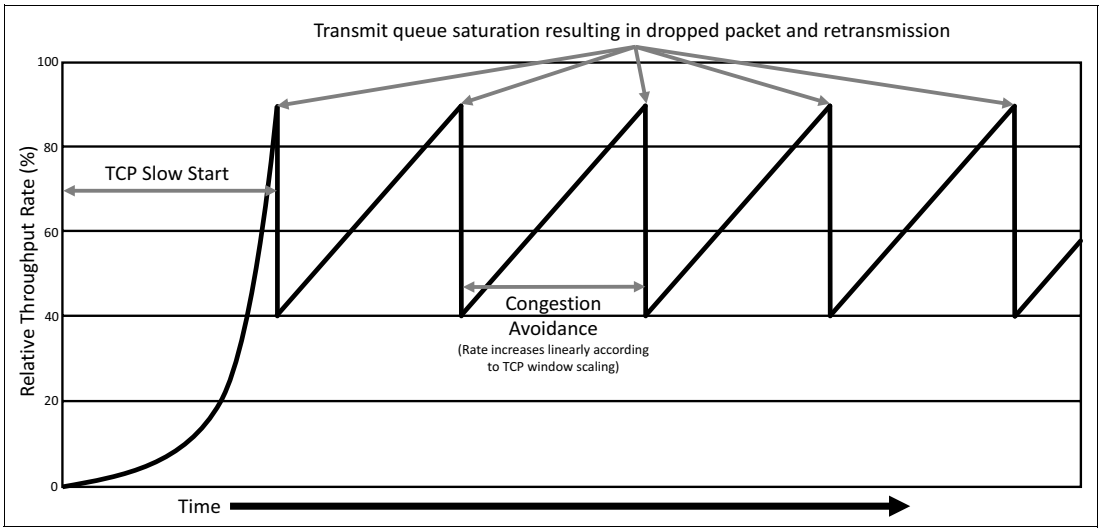


Figure 8-2 TCP saw tooth traffic pattern because of congestion

8.2.1 General Parallel File System

IBM General Parallel File System (GPFS) optimizes file access by splitting the file access across multiple nodes. When implemented over Ethernet, the connections are accomplished by using multiple TCP connections. The use of more than one NIC in an active-active configuration rather than active-standby or transmit load balancing (TLB) is preferred for redundancy and for increasing the bandwidth. Active-active NICs are accomplished by using link aggregation.

The use of VLAG enhances the reliability because the LAG is split across two switches. VLAG can also increase the performance because more connections between multiple switch hops can be used. Because there are multiple TCP connections, it is preferred to add Layer 4 ports into the trunk hash calculations by using the **portchannel thash l4port** configuration command.

Because GPFS is implemented by using TCP, enabling CEE on all nodes can improve the overall performance by reducing the network congestion.

It is important to also verify that the links are not over-used, which results in dropped traffic. These links can be monitored by monitoring the interface counters and looking for IBP/CBP and HOL-blocking discards. It is also good to review the unicast packets that are being transmitted and received across LAGs to verify that a good distribution is achieved. Egress distribution is controlled on the local device, whereas ingress distribution is controlled on the remotely attached device. It is useful to clear the counters by using clear interfaces and monitor them at intervals.

8.2.2 Internet Small Computer System Interface

Internet Small Computer System Interface (iSCSI) is another Ethernet SAN that is implemented by using TCP similar to GPFS, except that the file access is accomplished by using a single-server interface. Because a single file server is used, the use of LAGs is more critical for the server connections. Implementing the network by using VLAG is important for active use of all connections and enabling Layer 4 port into the trunk hash calculations might increase even distribution across the links (**portchannel thash l4port**).

The use of CEE improves the performance if all of the switches support CEE. There also are CEE enhancements available for iSCSI by using Data Center Bridge Exchange (DCBX) protocol. This configuration is similar to the DCBX protocol enhancements that are available for FCoE in which optimal connections are negotiated through the network. For an iSCSI network, CEE and iSCSI DCBX support must be enabled by using the following commands:

```
configure terminal
cee enable
cee iscsi enable
```

It is important to monitor the port counters to verify even distribution and congestion control. The use of 40 Gbps network connections can be helpful in optimizing an iSCSI network design because multiple TCP connections are not used; therefore, LAGs might not have effective traffic distribution. Also, 40 Gbps links reduce the transmission latency over the links.

8.2.3 Remote Direct Memory Access Over Converged Ethernet

Remote Direct Memory Access Over Converged Ethernet (RoCE) performance optimization is similar to SAN optimization. Latency is critical to RoCE; therefore, it is important to limit the number of hops through which the traffic passes. As the protocol name indicates, CEE is used; therefore, it must be enabled by using the **cee enable** command.

9

Integrating with hosts

This chapter provides information about integrating Lenovo switches with host operating systems.

This chapter includes the following topics:

9.1 Introduction to integrating with hosts

Although each operating system has various specific requirements, there are some items that are common between all (or at least most) operating systems. Consider following points when you are attaching any server to a Lenovo switch:

- Because servers rarely (if ever) should participate in Spanning Tree, it is important to add the **spanning-tree portfast** command (**spanning-tree edge** on older code) to any port that is facing a server. This configuration ensures that when the port comes up, it can immediately start forwarding packets (without this option set, the port discards packets for 30 seconds, which delays the time it takes for a server to become available, and worse can break things, such as DHCP or PXE boot for that host, which usually happens after the link is brought up, but often fails if portfast/edge is not configured on the port).

  For most Lenovo switches, the portfast/edge command does not take effect until the link is brought down and up at least one time after the command is enabled.

  You can determine whether the portfast/edge command took effect by running the **show span** command, and looking on the far right of the output for that port or virtual local area network (VLAN) in question for the keyword *edge*. If the keyword edge is not present in the output on the far right for that port, the command is not on the port, the command was added to the port but the link did not flap at least once, or the port is receiving Bridge Protocol Data Unit (BPDU) packets (a port that receives BPDUs never goes into edge mode to prevent loops).

  Not having portfast/edge on a server-facing port can also cause that port to stop forwarding packets for 30 seconds when *other* server ports flap their link in the same VLAN. This issue occurs because when a port comes up and it is not receiving BPDUs (servers should normally not be sending BPDUs), the switch does not know whether that port is going to be part of a loop; therefore, any other port that is also not receiving BPDUs (that does not have the portfast/edge command in effect) also stops forwarding packets for 30 seconds to ensure that a loop was not introduced when that link came up. To avoid this major disruption to an L2 network, *always* add the portfast/edge option to *all* server-facing ports.

- Another Spanning Tree best practice is to ensure network stability from a server that was incorrectly configuring the network interface cards (NICs) to attempt to participate in Spanning Tree, which is rarely wanted. To protect a port from a Spanning Tree mis-configured host, the BPDU guard and Root guard features can be enabled on a per-port basis, and must be considered for all server facing ports.

  BPDU Guard err-disables a port if it receives a BPDU packet. This action helps prevent issues if cables are attached incorrectly or if an end host somehow bridged their NICs. Consider the following points:

  – Setting BPDU guard on a switch that has Spanning Tree disabled does not do anything (the BPDU guard is not functional if Spanning Tree is disabled).

  – For BPDU guard to become effective, the port must also be configured with by using the **spanning-tree portfast** (**spanning-tree edge** on older code) command on Lenovo switches.

  Root Guard on a server-facing port helps to improve network stability by err-disabling a port if that port attempts to announce a better path to the root when that path should *never* be pointing toward the root. Root guard often must be placed on all ports that are facing away from the expected or wanted root path.

- If the server is using untagged packets only, set that server-facing port as untagged (**`switchport mode access`** in newer code) for maximum security. If tagging is used by the connecting host, the proper native/pvid (untagged) VLAN must be set on that port and *only* the VLANs that the server needs must be allowed on the port that is facing that server. This configuration increases the security and performance of the link.

  Security is improved because the server cannot start using VLANs it was never meant to use.

  Performance is improved because bandwidth is not wanted on the link by carrying broadcast and multicast traffic down to the host on VLANs that the host is not using.

9.2 Integrating with VMware vSphere

Integrating VMware vSphere with System Networking products includes several key options and considerations from the NIC Teaming for identifying vSphere host physical port connectivity. In this section, we describe ideas, planning recommendations, preferred practices, and troubleshooting tips.

9.2.1 vSphere vNetwork options with a standard vSwitch

A standard vSwitch includes the following options that enable various traffic distribution methods:

- Route that is based on Originating Virtual Port:
  - Host side

    This teaming option is the default and the most commonly used option. It distributes traffic by pinning VMs to a physical uplink. For example, if there are two physical ports within a Port Group with 20 VMs allocated to the port group, the distribution often is an even split of 10 VMs that are pinned to the first physical uplink. The remaining 10 VMs are pinned to the second physical uplink.
  - Switch side

    This option requires no special setting on the switch ports.
- Route that is based on source MAC hash
  - Host side

    This option distributes traffic that is based on the source MAC address of the VM guest. This option also uses a form of pinning VM guests to a specific uplink that is based on the source MAC. As with the first option, this option provides an even distribution if the number of VM guests are greater than the number of physical uplinks.
  - Switch side

    This option requires no special setting on the switch ports.

- Route that is based on IP hash
  - Host side

    This option distributed traffic that is based on source and destination IP hash across all active uplinks within the Port Group. IP hash load balancing should be set for all port groups that use the same set of uplinks. In VMware vSphere 5.0 and above, this option uses a static form of Port Aggregation that also is known as an Etherchannel and *not* Link Aggregation Control Protocol (LACP). LACP is supported only when a Distributed Virtual Switch is configured; however, the option selection is the same within vSphere when Static or Dynamic (LACP) PortChannels are configured. The difference is where the commands to implement LACP are applied; that is, the settings are in the vSwitch on the standard vSwitch versus on the distributed vSwitch where they are in the *Teaming and failover* section of the PortGroup.
  - Switch side

    When IP hash on the vSphere host side is used, the directly connect switch ports must form a static link aggregation. If you are connecting to multiple physical switches, the switches must form a stack or Virtual Link Aggregation (vLAG) to be seen as a single virtual switch.

    Use the following command to configure ports 21 and 22 in a PortChannel:

    ```
    portchannel 1 port 21, 22 enable
    ```
- Use explicit failover order
  - Host side

    The use of this option provides a single active uplink for all guest VMs that are using this port group. The remaining uplinks within the standby adapters list are used only if the active adapter failed. After the failed primary adapter recovered from loss of link “Link status only” or its gateway recovered “Beacon probing”, the VM guests are moved back to the primary “Active” adapter.
  - Switch side

    This option requires no special setting on the switch ports.

9.2.2 vSphere vNetwork options with a distributed vSwitch

Distributed vSwitch creates a single PortChannel across multiple vSphere hosts that use their available uplinks. A distributed vSwitch has several options that enable various ways of traffic distribution. VMware vSphere calls these port bindings, of which each type features various options.

Route based on originating virtual port, Route based on source MAC hash, and Use explicit failover order on a Distributed vSwitch have the same requirements as a standard vSwitch setup. However, the following options are extra selections or have different requirements:

- Route based on IP hash:
  - Host side

    This option distributes traffic that is based on source and destination IP hash across all active uplinks within the Port Group. IP hash load balancing should be set for all port groups that use the same set of uplinks.

    Route based on IP hash uses different PortChannel mechanism of bonding between a standard and distributed vSwitch. The configurations on the host side are identical; however, it is the directly connected switch that requires a change.

When this option is configured within a distributed vSwitch, the directly connected switch must have LACP configured for the host to communicate with the network. LACP uses a protocol LACPDU (data unit) to communicate and acknowledge that the same set of ports on either side are properly configured and added to the same LACP PortChannel. If ports within the same switch that are connecting to the same vSphere host are assigned to different LACP PortChannels or the host was misconfigured, these issues can cause instability or complete loss of connectivity to the network.

Figure 9-1 shows initial configuration settings when LACP is configured in the vSphere Web Client. The default load balancing setting, source and destination IP addresses, TCP/UDP port, and VLAN, is preferred because it provides for the best distribution of VM Guests across the ports within the PortChannel. However, in some cases, another form of load balancing might be required.

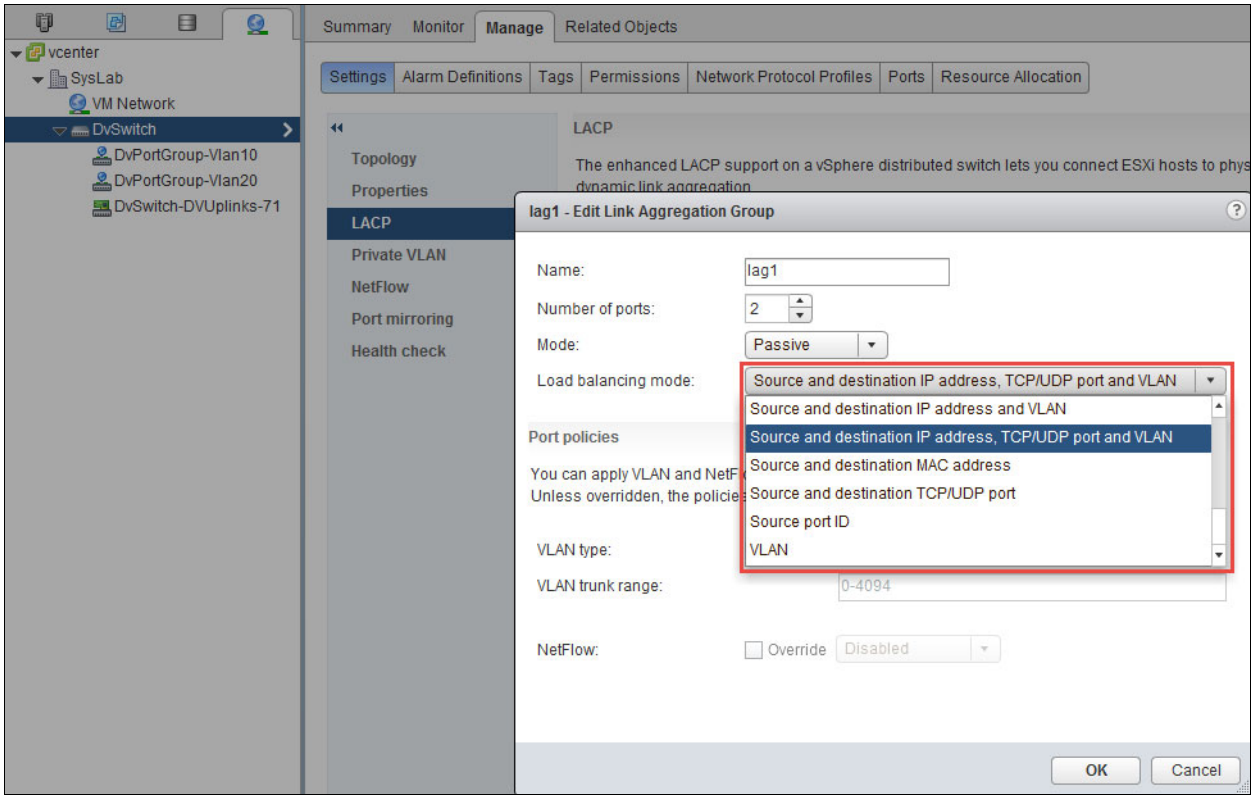


Figure 9-1 vSphere Web Client Initial LACP settings

Figure 9-2 shows setting up a new Distribution PortGroup. Each PortGroup that is participating in the LACP PortChannel must have Advanced option selected.

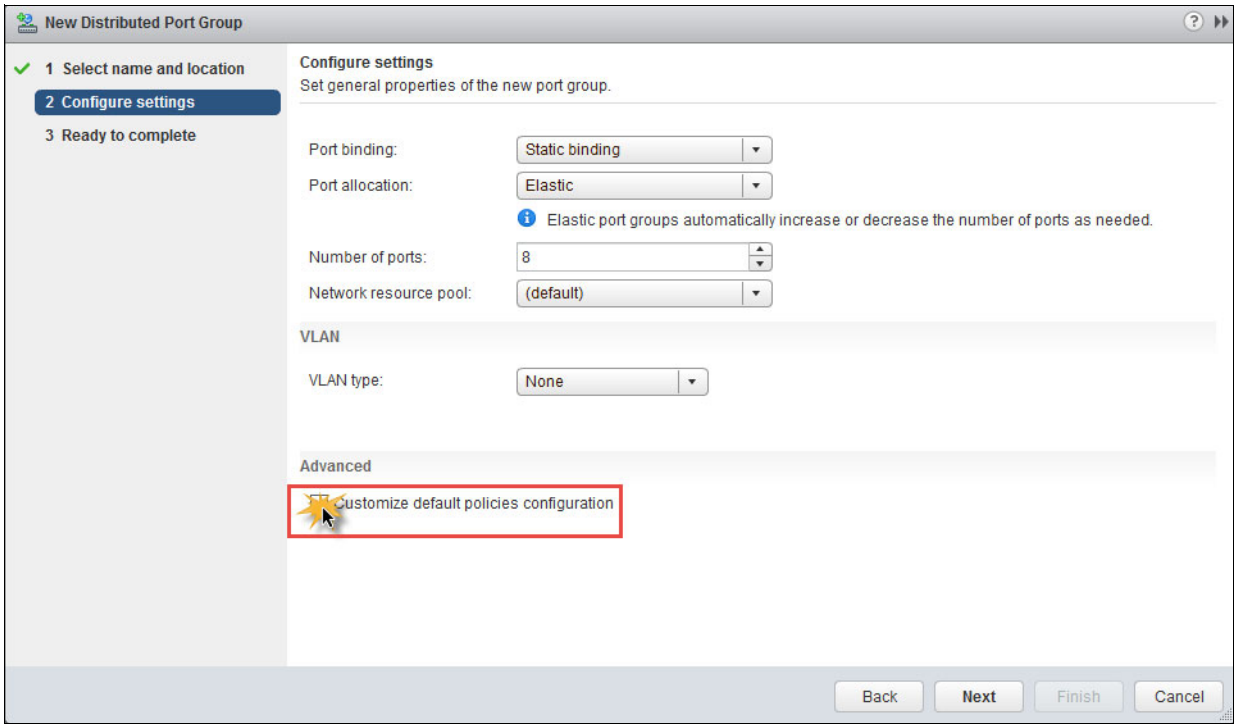


Figure 9-2 vSphere Web Client Advanced selection

Figure 9-3 shows removing the physical ports from the Port Group and adding only the lag1 LACP PortChannel. When a link aggregation group (LAG) is selected as the only active uplink, the load balancing mode of the LAG overrides the load balancing mode of the port group.

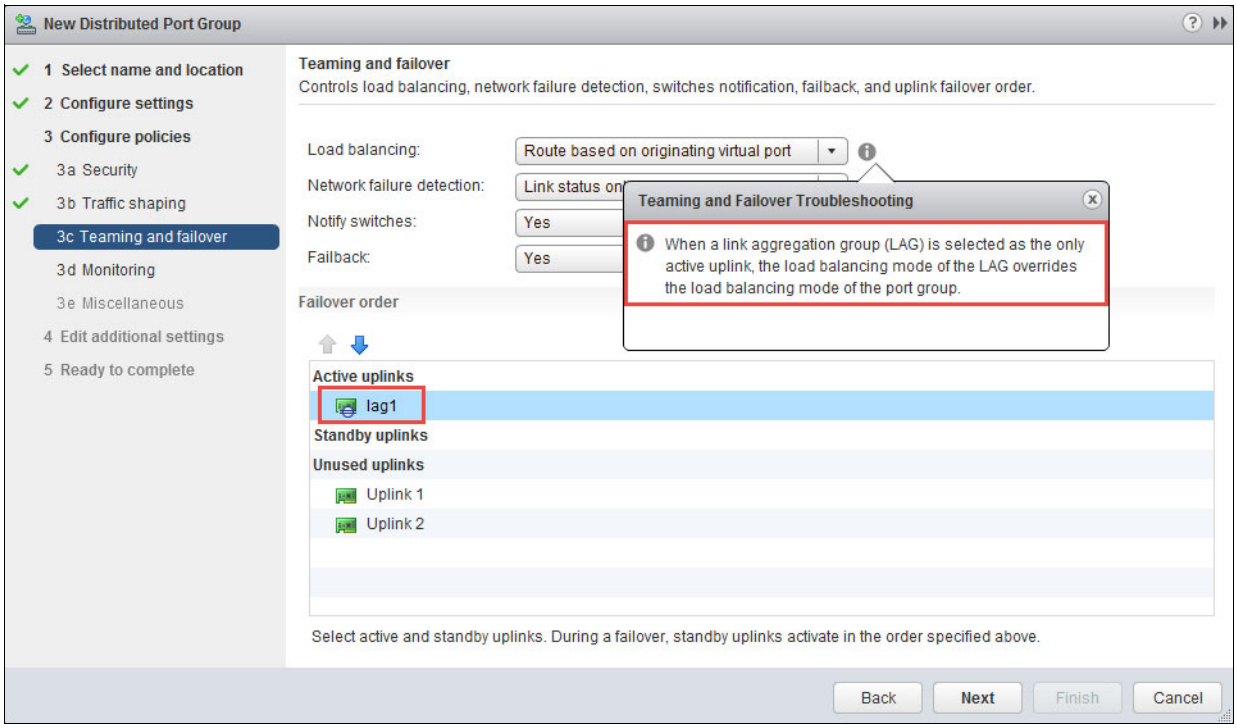


Figure 9-3 vSphere Web Client teaming and failover

- Switch side: When IP hash on the vSphere host side is used, the directly connect switch ports must form a dynamic LACP link aggregation. If you are connecting to multiple physical switches, the switches must form a stack or vLAG to be seen as a single virtual switch.

  Example 9-1 shows the configuration portion of a G8264 that is configured with LACP on ports 17 and 18 that are set to mode active with a unique administrators key.

Example 9-1 LACP PortChannel (that is mode active)

```
interface port 17
        lacp mode active
        lacp key 1718
!
interface port 18
        lacp mode active
        lacp key 1718
```

9.2.3 Identifying port connectivity from a vSphere Distributed vSwitch by using LLDP

Link Layer Discovery Protocol (LLDP) is a powerful tool for troubleshooting switch-to-switch connectivity during initial installations or when new switch hardware is added to an environment. As with Cisco Discovery Protocol (CDP), LLPD can send and receive information about the physical surroundings, such as remote port ID, remote host name, and remote IP address. However, a vSphere host also can display and transmit information to the attached switch that supports LLPD.

Consider the following points regarding vSphere host displaying and transmitting its physical connectivity information and reporting it to the attached switch:

- LLDP is supported on VMware vSphere 5.0 and above only.
- This process works with Distributed Virtual Switches (DVS) only.
- By default, a vSphere host is configured to receive information from a directly connected switch only. For the switch to see information from the vSphere host, it must be configured to transmit LLDP information.
- The physical NIC within the vSphere Host must support LLDP.

Most (if not all) of these conditions are standard in today's 1 Gb and 10 Gb data center environments.

Figure 9-4 shows the Type and Operation of an LDDP configuration within a Distributed vSwitch. The Type and Operation must be set accordingly for the information to be received and transmitted by the vSphere host NIC.

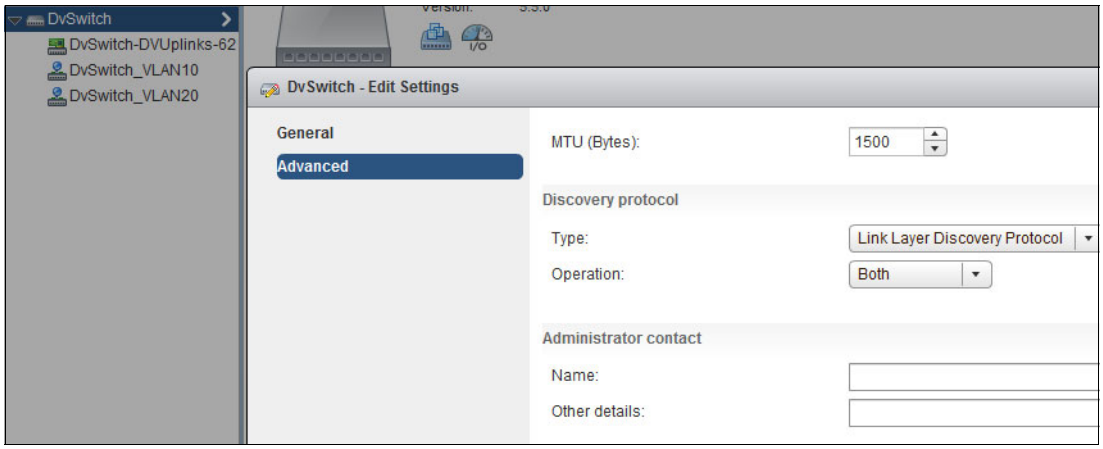


Figure 9-4 LLDP Type and Operational state

Figure 9-5 shows the results after LLDP is enabled on a selected distributed vSwitch.

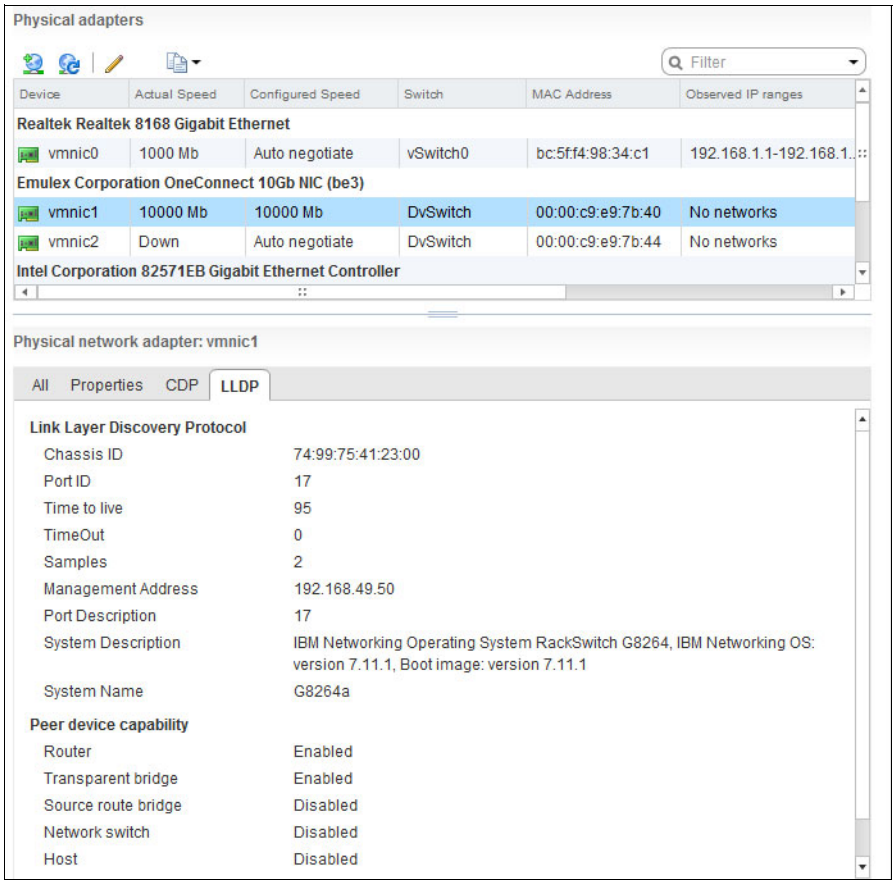


Figure 9-5 LLDP enabled information transmitted by the attached switch

9.2.4 Identifying port connectivity by using LLDP

LLDP is better known as a protocol that is used between switches. However, with LLDP enabled on a VMware vSphere host and the connecting switch (both of which are transmitting and receiving LLDP information), system administrators and network administrators know where things are within a data center.

Example 9-2 on page 202 shows the MAC address of a vSphere host and the host name, IP address, vmnic ID, and part number of an installed NIC.

Example 9-2 Output of a **s***how lldp remote port command with lldp enabled on a vSphere host*

```
G8264a#show lldp remote port
LLDP Remote Devices Information
Legend(possible values in DMAC column) :
NB   - Nearest Bridge          - 01-80-C2-00-00-0E
NnTB - Nearest non-TPMR Bridge - 01-80-C2-00-00-03
NCB  - Nearest Customer Bridge - 01-80-C2-00-00-00
Total number of current entries: 8

LocalPort | Index | Remote Chassis ID          | Remote Port          | Remote System Name                | DMAC
----------|-------|----------------------------|----------------------|-----------------------------------|---------
1         | 2     | 74 99 75 41 3b 00          | 1                    | G8264b                            | NB
1         | 3     | 74 99 75 41 3b 00          | 1                    |                                   | NnTB
5         | 5     | 74 99 75 41 3b 00          | 5                    | G8264b                            | NB
5         | 6     | 74 99 75 41 3b 00          | 5                    |                                   | NnTB
17        | 4     | 00 00 c9 e9 7b 41          | 00-00-c9-e9-7b-41    | 95Y3752 FWVer:10.2.370.15         | NB
17        | 8     | vmnic1                     | 00-50-56-59-7b-40    | esxi111.home.loc                  | NB
MGT       | 1     | 74 99 75 41 3b 00          | 65                   | G8264b                            | NB
MGT       | 7     | 00 25 03 19 d3 00          | 44                   | G8000                             | NB
```

9.3 Integration with VIOS

The Virtual I/O Server (VIOS) facilitates the sharing of physical I/O resources among client logical partitions within an IBM Power Systems server. VIOS presents the network to the IBM PowerVM virtual machine (VM) or Logical Partition (LPAR) by using the following methods:

- Bridged networking
- Open vSwitch
- User mode networking
- NAT networking
- PCI pass-through

Because most implementations use the bridged networking or Open vSwitch methods, only these methods are described in this section.

9.3.1 Bridged networking

Bridged networking provided the most efficient sharing of the network resources between VMs before Open vSwitch support on Power8 was available, which provides equivalent performance. The bridged networking method connects the physical LAN (Ethernet adapter or link aggregation) to a virtual Ethernet network by creating a virtual Ethernet bridge, as shown in Figure 9-6 on page 203.

This method implements a network bridge as described by the IEEE 802.1D standard. This bridge passes Layer 2 packets so Layer 3 packets are forwarded transparently.

The hypervisor is a virtual switch that provides the virtual Ethernet support to the VMs and enables communication between VMs without accessing the physical network. The PowerVM Virtual Ethernet includes the following major features:

- The virtual Ethernet adapters can be used for IPv4 and IPv6 communication and can transmit packets with a size up to 65,408 bytes. Therefore, the largest maximum transmission unit (MTU) for the corresponding interface can be 65,394 (or 65,390 if VLAN tagging is used).
- The IBM POWER Hypervisor presents to partitions as a virtual 802.1Q-compliant switch. The maximum number of VLANs is 4096. Virtual Ethernet adapters can be configured as untagged or tagged (following the IEEE 802.1Q VLAN standard).

- A partition can support 256 virtual Ethernet adapters. In addition to a default port VLAN ID, the number of extra VLAN ID values that can be assigned per virtual Ethernet adapter is 20, which implies that each virtual Ethernet adapter can be used to access 21 virtual networks.
- Each partition operating system detects the VLAN switch as an Ethernet adapter without the physical link properties and asynchronous data transmit operations.

Figure 9-6 shows the bridge architecture.

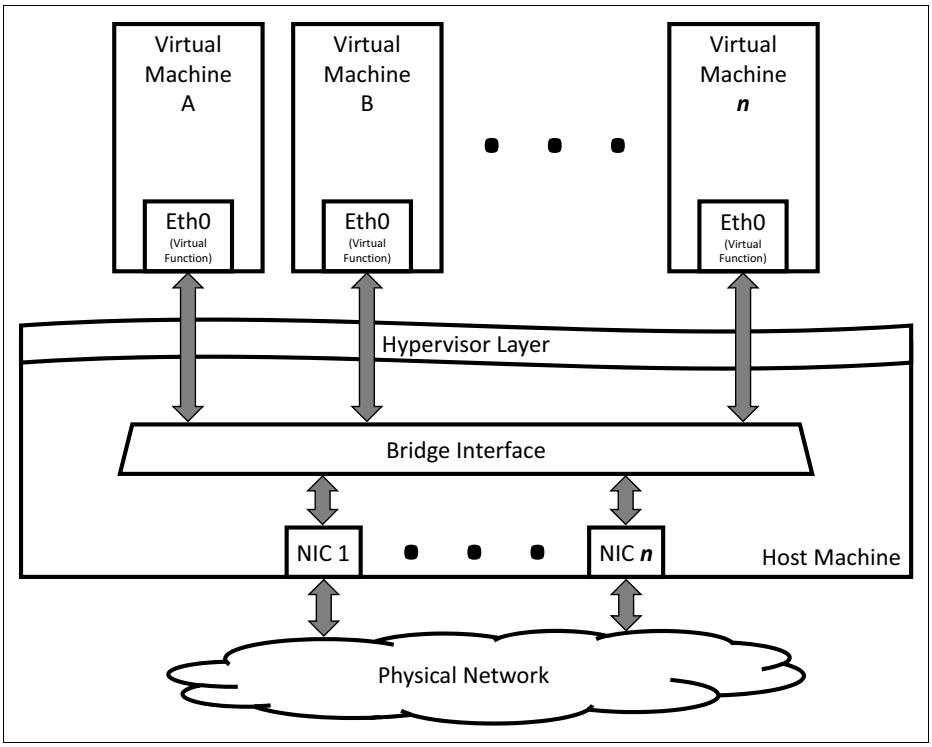


Figure 9-6 Bridge architecture

The virtual Ethernet adapters have connectivity outside of the server by using the bridge interface to a physical Ethernet adapter, which is also known as *Shared Ethernet Adapter*. This adapter can connect to the physical Ethernet adapter as a single NIC, Network Interface Backup, or by using Link Aggregation (Static or LACP). If Link Aggregation is used, it also must be configured on the Lenovo Networking switch. The switch also must be configured to carry all of the VLANs that are configured in VIOS.

It is considered a best practice to provide network redundancy through more than one network switch. On the Lenovo Networking switch, it is recommended to use VLAG and connect the switches to NICs by using a single link from each switch or (if possible) a full mesh configuration to four NICs. These NICs must be configured as a PortChannel or LACP aggregation.

9.3.2 Open vSwitch

Power8 enables support of Open vSwitch, which is an open source, multilayer distributed virtual switch. It supports standard management protocols and interfaces, such as NetFlow, sFlow, SPAN, LACP, and 802.1ag. The key advantage of Power8 is that Open vSwitch can be distributed across many physical machines. By providing APIs for automation, Open vSwitch becomes an almost perfect solution for OpenStack deployments.

Open vSwitch replaces the VIOS hypervisor and bridge interfaces.

9.4 Integrating with Linux hosts

This section describes options for configuring a Linux host when it is connected to Lenovo switches. It primarily describes a scenario in which the host is running on a compute node in a Pure Flex chassis. However, the configuration that is presented can be adapted for use in BladeCenter chassis or to use a pair of rack-mounted switches and dual-homing the hosts to them.

The examples in the following sections focus on servers with two physical NICs, but they can be extended to designs where four, six, or more NICs are provisioned on a server. Such servers often use the available NICs in pairs and attach each pair to a different part of the network. The techniques that are shown in the examples that follow allow each pair to provide high availability and redundancy.

9.4.1 Switch configuration options for Linux hosts

A host can be configured when it is connected to two switches by using one of the following methods:

- Aggregating the NICs that connect to the switches together into a PortChannel.
- Configuring failover so that if the primary (active) switch fails, the bonding driver on the server redirects traffic to use a NIC that is connected to the backup switch.

For these options, the Linux bonding driver is used. This driver aggregates multiple NIC ports into a single logical NIC as seen by the Linux operating system; these ports often are called `bond<x>`.

Figure 9-7 shows sample switch configuration commands for each of these options.

```
interface port 3:INTA1,4:INTA1
lacp mode active
lacp key 1001
```

Figure 9-7 Switch configuration to aggregate host ports by using stacking or Flex Fabric: LACP

The configuration that is shown in Figure 9-7 forms an LACP PortChannel out of two ports on two physical switches in a Flex System chassis that support a single server. Similar configurations can be used in a stacked configuration: in a BladeCenter chassis or with rack-mounted switches and servers. This configuration requires that the Linux bonding driver is configured to use LACP; that is, in 802.1ad mode (mode=4), as shown in Figure 9-8.

```
portchannel 10 port 3:INTA1,4:INTA1 enable
```

Figure 9-8 Switch configuration: Static aggregation of host ports by using stacking or Flex Fabric

This configuration uses static link aggregation on the server-facing ports rather than by using LACP. The bonding driver can be configured as mode=0 (round-robin) or mode=2 (XOR of MAC addresses).

Figure 9-9 shows vLAG configuration with LACP.

```
interface port INTA1
lacp mode active
lacp key 1001
vlag adminkey 1001 ena
```

Figure 9-9 vLAG configuration with LACP

The configuration commands that are shown in Figure 9-9 can be used on each of a pair of switches by using vLAG to create a server-facing aggregation. On the server, this configuration is identical to the configuration that is shown in Figure 9-7 on page 204 and requires configuration of the bonding driver with mode=4. (The commands to establish the vLAG connection between this switch and its peer are not shown in Figure 9-9. vLAG is described in 2.1, “Sample topologies” on page 6).

Figure 9-10 shows vLAG configuration with static PortChannel.

```
portchannel 10 port INTA1 enable
vlag portchannel 10 enable
```

Figure 9-10 vLAG configuration with static PortChannel

The configuration commands that are shown in Figure 9-10 can be used similarly to the commands in the example that is shown in Figure 9-9, but they create a static aggregation that faces the server. For the server, this configuration is identical to the one that is shown in Figure 9-8 on page 204 and requires the same configuration on the bonding driver (mode=0 or mode=2).

In the template that is shown in Figure 9-11, the item that is monitored is intended to be a port or channel that connects to the upstream network. The item that is controlled is intended to be a port or channel that is connected to a server. In a Flex System or BladeCenter chassis, the item that is monitored is one or more external ports; the item that is controlled is one or more internal ports.

```
failover trigger 1 mmon monitor [member | adminkey | portchannel] <x>
failover trigger 1 mmon control [member | adminkey | portchannel | vmember] <x>
failover trigger 1 enable
failover enable
```

Figure 9-11 Switch configuration template for failover

The *failover* feature automatically brings the controlled ports down if all of the monitored ports fail. This process causes the bonding driver to fail over to a port that is directly connected to another switch and ideally can reach the upstream network via a different path. The bonding driver must be configured with mode=1 (active/standby) in this configuration; in normal operation, the active NIC carries all incoming and outgoing traffic from the server.

Preferred configuration

Where possible, one of the configurations that uses a server-facing aggregation is preferred. In most cases, other than stacking or Flex Fabric implementations, this configuration includes the use of the vLAG feature on two server access switches.

In some scenarios where a server has more than two NIC ports, you can aggregate all of the ports together similar to the configurations that are shown in Figure 9-7 on page 204 and Figure 9-8 on page 204. Alternatively, you can create multiple pairs of NICs where each pair can connect to a different part of a customer's network and be used for a different purpose (for example, different security zones, such as a DMZ).

The failover configuration is less preferred because it uses one NIC purely as a backup; the backup NIC sits idle during normal operations.

9.4.2 Configuring the bonding driver

Bonded interfaces, such as bond0, can be configured in several different ways, depending on whether they are to be made permanent (capable of surviving a reboot), and on which Linux distribution is being used. Figure 9-12 shows examples of these configurations.

```
ifconfig bond0 192.168.1.1 netmask 255.255.255.0
ifenslave bond0 eth0 eth1
```

Figure 9-12 Creating a bond interface

The commands that are shown in Figure 9-12 create a `bond0` interface and use `eth0` and `eth1` as slaves. However, the interface that is created by these commands does not survive a reboot. The **`ifconfig`** command and the **`ifenslave`** command are replaced by options of the **`ip`** command in newer versions of Linux. The `bond0` device that is created by the command that is shown in Figure 9-12 has a default setting to mode=0, which uses a round-robin approach to the use of the physical devices in active/active mode. This configuration is not the most preferred technique to configure a bond interface.

To create a permanent bond interface, the techniques vary depending on different releases and different Linux distributions. An example for Red Hat Linux and related distributions (Fedora, and so on) is shown in Figure 9-13.

```
DEVICE="bond0"
IPADDR=192.168.1.121
NETMASK=255.255.255.0
NETWORK=192.168.1.0
BROADCAST=192.168.1.255
ONBOOT=yes
BOOTPROTO=none
USERCTL=no
TYPE=Ethernet
MASTER=yes
BONDING_OPTS="bonding parameters separated by spaces - including mode"
```

Figure 9-13 Red Hat config file for the bond0 interface

Figure 9-14 shows a Red Hat configuration for an enslaved physical interface.

```
DEVICE="eth0"
ONBOOT=yes
USERCTL=no
TYPE=Ethernet
BOOTPROTO=none
DEFROUTE=yes
MASTER=bond0
SLAVE=yes
```

Figure 9-14 Red Hat configuration for an enslaved physical interface

A similar file is required for the `eth1` interface.

These files all have names of the form `ifcfg-<interface>`, such as `ifcfg-eth0` or `ifcfg-bond0` and are stored in the `/etc/sysconfig/network-scripts` directory.

Use the following options if they are available, which apply to most Linux distributions:

- The `miimon` options specifies a delay for failing over a failed member of a bond to avoid needless flapping (measured in milliseconds). Typical values are in the 100 - 200 range.
- The `primary` option selects which NIC is to be the active NIC when the bond is in *active/standby* mode (mode=1). It is not applicable in any other mode. An example of syntax is `primary=eth0`.
- The `xmit_hash_policy` option is the equivalent to the commands on the switches that manage allocating traffic to the physical links in a PortChannel or LACP channel. The default is to use L2 hashing, which uses the source and destination MAC addresses.

  The value 0 = layer2, 1 = layer3, and 2 = both.
- The `updelay` sets a delay for recognizing a link-up condition on a physical NIC to make sure that it does not go back down quickly.
- The `lacp-rate` default value is 0 (slow), which is consistent with the default setting for Lenovo switches.

9.4.3 Configuring Linux to support multiple VLANs on a NIC

A Linux host can be configured to send and receive traffic on multiple vLANs. This configuration is done by extending 802.1q tagging on the server-facing ports on the access switches and configuring the NIC. This configuration can be done by using commands that do not survive a system reboot and by editing appropriate configuration files so that the VLAN interface is permanent.

When a VLAN interface is created, it is managed by Linux as its own device. VLAN interfaces that are created to use `eth0` (`/dev/eth0`) are by default seen as `eth0.<vlan number>`. An example is `eth0.10` for VLAN 10. VLAN interfaces can also be created by using bonding master interfaces, such as `bond0`. For example, they can appear as `bond0.10`.

For typical NICs and bonding interfaces, the standard interface name continues to be available for sending and receiving untagged traffic. This untagged traffic is seen by the switches on the native VLAN of the server-facing port. If the server-facing port on the switch is configured to not handle untagged traffic (by using the **`vlan dot1q tag native`** command), this traffic is discarded.

Important: The Linux operating system also provides a mechanism to apply multiple IP addresses to a single NIC. The device names that are created by this mechanism appear by default in the form `eth0:<instance>`. These interfaces do not use 802.1q VLAN tagging.

VLAN interfaces are configured in the same ways as standard NIC interfaces. The **`vconfig`** command and the **`ip`** command can be used to assign addresses, as shown in Figure 9-15 and Figure 9-16.

Figure 9-15 shows the Linux commands that are used to create a VLAN interface.

```
vconfig add eth0 10
ifconfig eth0.10 192.168.1.102 netmask 255.255.255.0 up
-or-
ip link add link eth0 name eth0.10 type vlan id 10
ip addr add 192.168.1.102 dev eth0.10
ip link set dev eth0.10 up
```

Figure 9-15 Linux commands to create a vlan interface

VLAN devices can be made permanent by creating and editing configuration files in the same ways as is done for standard NIC devices. Red Hat distributions complete this process by manipulating files with names of the form `ifcfg-eth0.10` in the `/etc/sysconfig/network-scripts` directory. An example of such a file is shown in Figure 9-16.

```
DEVICE=eth0.10
BOOTPROTO=none
ONBOOT=yes
IPADDR=192.168.1.1
NETMASK=255.255.255.0
USERCTL=no
NETWORK=192.168.1.0
VLAN=yes
```

Figure 9-16 Example of the /etc/sysconfig/network-scripts/ifcfg-eth0.10 file

Preferred practice configurations

For a single VLAN, use a redundant pair of NICs that are bonded when there are multiple VLANs in use. The individual VLAN interfaces can then be built on `bond0` or a similar interface; for example, `bond0.10`.

An alternative to this approach is to use the UFP feature to create multiple interfaces from each physical port on the NIC. These interfaces can each carry multiple VLANs if so configured, and their bandwidth limits can be configured. However, UFP provides only four virtual interfaces (vports) per physical port, or three if a virtual HBA is provisioned for storage networking. This approach is described in 9.4.4, “UFP and vNIC considerations for Linux” on page 209.

9.4.4 UFP and vNIC considerations for Linux

UFP and vNIC present a two-port 10 Gb NIC to the operating system as up to eight devices, with up to four devices that are associated with each physical port. Linux does not prevent these virtual devices from being used with any of the options of the bonding driver, but active/active bonding modes that require port aggregation on the switch are not supported.

The server cannot determine when UFP is in use; it sees devices identified as `eth0` - `eth7` (`eth5` if virtual HBA is provisioned).

It is a good practice to configure the vports (or vNICs) in like pairs across the two physical ports. The failover option is supported when this configuration is done starting in firmware release 7.9 for UFP. An example configuration that uses this feature is shown in Figure 9-17. This configuration differs from previous failover configurations because the item that is controlled is identified as a vmember, which identifies it as a UFP vport instead of a port, PortChannel, or LACP key.

Typically all four vports on each switch are set to fail over to their counterparts on the other switch, except for FCoE vports. A similar configuration is used in a rack-mounted environment with dual-homed servers. Multiple uplink ports or aggregations are often monitored and multiple server-facing ports and vports are controlled in the same trigger. It is also possible to use multiple triggers as needed, as shown in Figure 9-17.

```
failover trigger 1 mmon monitor adminkey 1001
failover trigger 1 mmon control vmember INTA1.1
failover trigger 1 ena
failover ena
```

Figure 9-17 Failover configuration to control UFP vport

vNIC does not offer the same level of versatility as UFP. However, where it is in use, failover is configured in an analogous way, where each vNIC in a physical port is set to fail over to its counterpart on the other switch (also excluding FCoE). As for UFP, link aggregation is not supported by vNIC instances, but the server does not prevent configuration of active/active bonding modes.

Figure 9-18 shows a sample configuration of vNIC failover. This configuration is not similar to the configuration that is used with typical ports and with UFP.

```
vnic vnicgroup 2
member INTA1.1
key 1001
failover
ena
```

Figure 9-18 Failover configuration for vNIC

The failover option for vNIC is part of the configuration of a vnic group. It applies to all members of the group, meaning that when all of the uplink ports that are associated with LACP key 1001 (or with an individual port or PortChannel that is configured as the uplink for the group) are down, all of the vnic members of the group are disabled to trigger failover to their counterparts.

9.4.5 Considerations for Linux guest VMs

The topology of network connections for a guest VM is typically configured in the hypervisor instead of the guest. In many cases, a guest VM has one virtual NIC only as seen by the hypervisor, even though the hypervisor might have multiple physical NICs and these physical NICs can also be virtualized by using vNIC or UFP.

It is possible for a NIC on a guest VM to carry tagging and be configured with VLAN interfaces (such as `eth0.10`) as is the case for a physical server. This configuration is done in ESX by configuring the guest network interfaces in a port group configured for guest tagging, which is accomplished by setting the port group to VLAN 4095. If this configuration is done, the switch ports that are facing the hypervisor server must include membership in all of the VLANs that any guest on that host might use.

It is similarly possible for a guest VM to have more than one virtual NIC. For a guest, this configuration is not required for deployment of a highly available environment; high availability is configured on the hypervisor rather than on the guests. Multiple NICs on the guest often can enable the guest to connect to different zones in the overall network, such as a DMZ (a firewall configuration for securing local area networks) that is between an inner and outer firewall or a portion of the network that is dedicated to running backups of stored data.

9.4.6 Summary of preferred configurations

This section summarizes the following preferred configurations:

- Use two switches to connect to the server, including when you use embedded switches in a chassis and when you use rack-mounted servers and switches. The server then has two NICs (or multiple pairs of NICs). NIC ports are almost always used in pairs, including the bonding driver on Linux servers. Multiple pairs of switches might be used in some designs.
- Configuration options that allow both NICs to be active deliver more bandwidth to the server than configurations in which some NIC ports are used only for backup. This configuration means that the use of the failover feature is less preferred than options, such as Flex Fabric, stacking, and vLAG.
- The features that allow active/active bonding each have their benefits and constraints. Also, be aware that some of these options are not supported on some of the Lenovo switching products.
- Multiple VLANs can be implemented by using 802.1q tagging on the server NICs or by the use of NIC virtualization options, including vNIC and UFP. Although each of the available options has its benefits and constraints, it is more common to use NIC virtualization options than tagging within a Linux server.

9.5 Integrating with Windows Server operating systems

Integrating Lenovo switching with Windows based systems is relatively straight forward, with most of the complexity involving various teaming modes that are available in the different versions of Windows Server.

For Windows Server editions before 2012, Microsoft did not provide built-in teaming, but instead relied on third-party utilities from the various NIC vendors. Starting with Windows server 2012, Microsoft now offers built-in teaming, but users can still use third-party tools, if wanted.

For more information about the various teaming modes that are available for different versions of Microsoft Server platforms, see the Windows Server teaming section in *NIC Virtualization in Flex System Fabric Solutions*, SG24-8223, which is available at this website:

http://lenovopress.com/sg248223

One bit of confusion that was noted with Windows deployments is that it is possible to go into the View Network connections section of Windows, highlight multiple NICs, right-click them, and select **Bridge Connections**, which is *not* teaming. This selection turns the NICs into an L2 bridge connection and potential create loops in the network, which is rarely wanted.

Related publications

The publications that are listed in this section are considered particularly suitable for a more detailed discussion of the topics that are covered in this book.

Lenovo Press publications

The following Lenovo Press publications provide more information about the topics in this document. some publications that are referenced in this list might be available in softcopy only:

- *NIC Virtualization in Flex System Fabric Solutions*, SG24-8223*:*

  http://lenovopress.com/sg248223
- Lenovo Press Product Guide on RackSwitch top-of-rack switches:

  http://lenovopress.com/systemx/tor

You can search for, view, download, or order these documents and other Redbooks, Redpapers, Web Docs, draft, and other materials at this website:

http://lenovopress.com

Online resources

The following websites are also relevant as further information sources:

- Lenovo systems networking:

  http://www.lenovo.com/networking
- Lenovo Enterprise Networking community:

  https://forums.lenovo.com/t5/Enterprise-Networking/bd-p/nw01_eg
- Documentation portal:

  http://ibm.com/support/entry/portal/Documentation
- Flex System Information Center:

  http://pic.dhe.ibm.com/infocenter/flexsys/information/

(0.2”spine)
0.17”<->0.473”
90<->249 pages

lenovo

Lenovo Networking Best Practices for Configuration and Installation

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

At Lenovo Press, we bring together experts to produce technical publications around topics of importance to you, providing information and best practices for using Lenovo products and solutions to solve IT challenges.

Benefit from the expansive knowledge of Lenovo Networking experts

Discover design strategies to maximize network performance

Learn about the latest switching and routing features

Implement switch security and management features

Networking is the foundation of any multisystem solution and this document provides you with the tools to enable a successful implementation. The features that are available on the Lenovo Networking switches can be overwhelming at first; however, you can construct it easily if the configuration is broken down into its basic building blocks.

Each chapter in this book focuses on the key areas of the switch configuration, starting with the network design and proceeding through the network layer features, security, and server integration. Details are provided about how to configure the switch and to assure that it is functioning properly. This publication does not provide a complete description of every switch feature but instead focuses on the topics that are generally considered key to a successful implementation.

This publication is targeted towards technical professionals (networking team, consultants, technical support staff, and IT specialists) who are responsible for supporting the integration of systems into new and existing networks.