ArubaOS-CX Virtual Switching Extension (VSX) Guide for 10.01

a Hewlett Packard Enterprise company

Part Number: 5200-5202
Published: July 2018
Edition: 1

Notices

The information contained herein is subject to change without notice. The only warranties for Hewlett Packard Enterprise products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Hewlett Packard Enterprise shall not be liable for technical or editorial errors or omissions contained herein.

Confidential computer software. Valid license from Hewlett Packard Enterprise required for possession, use, or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Links to third-party websites take you outside the Hewlett Packard Enterprise website. Hewlett Packard Enterprise has no control over and is not responsible for information outside the Hewlett Packard Enterprise website.

Acknowledgments

Intel®, Itanium®, Pentium®, Xeon®, Intel Inside®, and the Intel Inside logo are trademarks of Intel Corporation in the U.S. and other countries.

Microsoft® and Windows® are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Adobe® and Acrobat® are trademarks of Adobe Systems Incorporated.

Java® and Oracle® are registered trademarks of Oracle and/or its affiliates.

UNIX® is a registered trademark of The Open Group.

Applicable products

This document applies to the following products:

- Aruba 8400 Switch Series (JL375A, JL376A)
- Aruba 8320 Switch Series (JL479A, JL579A, JL581A)

Latest version available online

Updates to this document can occur after initial publication. For the latest versions of product documentation, see the links provided in the Websites chapter of this document.

About the examples

Examples in this document are representative and might not match your particular switch or environment.

The slot and port numbers in this document are for illustration only and might be unavailable on your device.

The software notation for describing module, slot, port, and interface information depends on the switch hardware.

Unless otherwise noted, examples in this document are based on a switch that identifies line module interfaces using member/slot/port notation, such as 1/1/1.

Switch prompts in examples

The switch prompts used in this document are examples and might not match your particular switch or environment.

In examples:

- The switch prompt starts with the word `switch`.
- The switch prompt also indicates the command context.

  For example:

  switch>

  Indicates the operator command context.

  switch#

  Indicates the manager command context.

  switch(config)#

  Indicates the global configuration context.

  In your environment, the switch prompt can vary because the prompt is user-configurable.

- Typically, the switch prompt begins with the host name of the switch.
- The switch prompt contains specifiers in certain configuration command contexts, such as interface name or VLAN ID. For example: `switch(config-vlan-100)#`

  In these cases, examples in this document might contain placeholders such as `n` or `if`.

This chapter provides information about upgrading customer configurations that include the MCLAG feature operating on ArubaOS-CX 10.00 to the replacement feature, Virtual Switching Extension (VSX), available in this release. VSX provides advanced capabilities beyond what was previously available with the MCLAG feature. This chapter provides the procedure for upgrading to VSX and it describes configuration differences and equivalencies between MCLAG and VSX.

During the upgrade, the MCLAG configuration is removed and VSX must be configured through the CLI. The upgrade process does not automatically migrate the configurations that include MCLAG on ArubaOS-CX 10.00 to VSX in this release.

NOTE: A maintenance window is required for this upgrade. Perform the upgrade with the assistance of support engineers for minimizing network downtime.

What happens during the upgrade

After the switch boots the software for this release, the MCLAG-related configuration is removed allowing for the configuration of VSX. The following configuration changes occur:

- All MCLAG configuration items are removed, which includes the following:
  - The MCLAG ISL (Inter-Switch Link)
  - The MCLAG keepalive
  - Any MCLAG keepalive hello, hold, or dead timers
  - The MCLAG device priority
- All active-gateway configuration items are removed
- All multichassis LAG configurations remain configured
- All multichassis LAGs are held in a down state until VSX is configured

Upgrade procedure overview

There are two procedures for upgrading from ArubaOS-CX 10.00 to this release for the transition from MCLAG to VSX; each procedure is useful for different network environments.

NOTE: Follow the upgrade procedure for your situation within a maintenance window because network availability cannot be guaranteed. Plan to have at least a two-hour maintenance window in case problems arise. Some of the preparation can be done before the upgrade to minimize the downtime.

Rolling upgrade for switches

To minimize the amount of traffic down on a network, the rolling upgrade procedure involves upgrading each of the switches serially. This procedure takes longer overall but it ensures the least amount of downtime for the traffic on the network. The rolling upgrade procedure is useful for network environments that require live traffic running during their maintenance window.

The process first involves the upgrade of one switch. Second, a subsequent transition of the traffic is moved over to that upgraded switch. Third, the remaining switch is upgraded. While the first switch is upgraded to this release, the other switch running version 10.00 still uses MCLAG until the VSX configurations are made to the first switch. Once the VSX configuration is applied to the first switch, the downstream MCLAG interfaces become active approximately 10 minutes after the configuration is applied. The remaining device can also now be upgraded to this release.

Upgrading both switches at the same time

The procedure for upgrading both switches at the same time is useful when you want the shortest amount of overall time to upgrade. This procedure is also useful when you can afford to have both switches offline during their maintenance window. This procedure uploads the software on both switches and boots them both into the new software at the same time. This procedure lets both switches boot in parallel, which saves time.

NOTE: Network traffic will NOT be forwarded across the switches until VSX is configured on the upgraded switches.

Performing a rolling upgrade

Prerequisites

1. Ensure that you have a scheduled maintenance window. This procedure is NOT hitless, but it does aim to minimize the amount of traffic loss. It is advisable to have a support engineer from HPE available during this procedure.
2. Enter the `show loop-protect` command for verifying that the action on loop detection has a value of `TX disable` on the MCLAG interface. If the setting has a different value, reset the value to `TX disable` by entering the `loop-protect action tx-disable` command:

   ```
   switch# configure terminal
   switch(config)# interface lag 2 multi-chassis
   switch(config-if)# loop-protect action tx-disable
   ```

3. Back up the current software to the secondary image bank. Run the `show version` command. If the `Active Image` is `primary`, copy the primary image to the secondary image bank:

   ```
   switch# copy primary secondary
   ```

   If the current image bank is secondary, the current 10.00 image is already in the secondary bank.

4. Save a copy of the running configuration on both devices:

   ```
   switch# copy running-config sftp://<username>@<IP>/filename vrf mgmt
   ```

5. Examine the 10.00 configuration files. Determine the appropriate VSX commands that will need to be run after the software on the device is upgraded. Many MCLAG commands have changed in this release. See **MCLAG to VSX command equivalency** on page 12 for finding the equivalent VSX commands.
6. Check the status of the `show mclag brief` command and validate the ISL is in-sync and the keepalive is established. Also, verify which switch is the primary switch. This procedure requires you upgrade the primary switch first. If the secondary switch is upgraded first, the downstream MCLAG interfaces will be stuck in LACP-block until both switches are upgraded to this release and the VSX configurations are applied.
7. Check the output of the `show lacp interfaces` command and note which LACP interfaces are in a forwarding state of up.

Procedure

1. Ensure that you are in a maintenance window. There will be some amount of network disruption even with the upgrade of a single switch.
2. Copy the software from this release onto both switches by using TFTP, SFTP, or USB:

```
switch# copy sftp://<username>@<ip>/XL_10_01_0001.swi primary vrf mgmt
```

3. Remove the keepalive configurations from both switches so that they can both forward traffic during the upgrade process. The keepalive is restored later when the VSX configuration is applied.

   a. Run the command `show mclag brief` for determining if the switch is the primary switch or the secondary switch.

   b. Run the following command on the secondary switch for removing the keepalive configuration:

```
switch(config)# no mclag keepalive
```

4. If the 10.00 MCLAG configuration uses active gateway instead of VRRP, shutdown on the MCLAG primary switch any ROPs, SVIs, or layer 3 LAGs that provide an uplink to the rest of the network. These uplinks would typically be links that are connected through OSPF, BGP, or static routes. This shutdown helps to prevent traffic from being routed to this switch until it is properly configured with VSX.
5. On the MCLAG primary switch, boot the switch to this software release:

```
switch# boot system primary
```

   While the primary switch is being upgraded and converted to VSX, the secondary switch is still actively forwarding traffic with MCLAGs from version 10.00 of the software. When you are prompted, save the configuration.

6. Once the primary switch has been booted for this release, all MCLAG configurations are removed with the exception of the MCLAG interfaces. Each of the MCLAG interfaces is in an LACP-block state until VSX is configured and activated.
7. With the information you learned about your configuration in step **5** in prerequisites, configure VSX on the switch by entering the `role primary` command:

```
switch(config)# vsx
switch(config-vsx)# role primary
switch(config-vsx)# inter-switch-link 1/1/1
switch(config-vsx)# keepalive peer 10.0.0.1 source 10.0.0.2
switch(config-vsx)# exit
switch(config)# interface vlan 10
switch(config-if-vlan)# active-gateway ip 10.1.1.1 mac 00:00:00:00:00:011
```

8. Save the configuration from the running configuration to the startup configuration by entering the `write memory` command.
9. Approximately 10 minutes after applying the VSX configuration, the VSX LAG interfaces will move from LACP-block to ACTIVE on the designated primary. Monitor the output of `show lacp interfaces` for determining when the interfaces are active.
10. Enable any ROPs, SVIs, or layer 3 LAGs that were previously disabled to the uplinks. The VSX configuration is applied, the upgrade is complete, and the primary switch is ready to forward network traffic.
11. On the designated secondary switch, boot the switch to this software release:

```
switch# boot system primary
```

The MCLAG configurations on the secondary switch are removed with the exception of the MCLAG interfaces.

12. With the information you learned about your configuration in step **5** in prerequisites, configure the VSX on the switch by entering the `role secondary` command:

```
switch(config)# vsx
switch(config-vsx)# role secondary
switch(config-vsx)# inter-switch-link 1/1/1
switch(config-vsx)# keepalive peer 10.0.0.1 source 10.0.0.2
switch(config-vsx)# exit
switch(config)# interface vlan 10
switch(config-if-vlan)# active-gateway ip 10.1.1.1 mac 00:00:00:00:00:011
```

13. Save the configuration from the running configuration to the startup configuration by entering the `write memory` command.
14. The MCLAG conversion to VSX is complete. Run the `show vsx brief` command on both switches.

    Verify that the ISL state is `In-Sync` and the keepalive state is `Keepalive-Established`.
15. Check the output of the `show lacp interfaces` command. Validate that the VSX LAGs have a forwarding state of up.
16. Configure VSX-related features as needed. Consult this guide for additional information about VSX features, such as VSX configuration synchronization and active forwarding.
17. Save the running configuration to the startup configuration:

```
switch# write memory
Success
```

Upgrading both switches at the same time

Prerequisites

1. Ensure that you have a scheduled maintenance window. There will be a disruption of service until the upgrade is complete and VSX is configured on the switches. It is advisable to have a support engineer from HPE available during this procedure.
2. Enter the `show loop-protect` command for verifying that the action on loop detection has a value of `TX disable` on the MCLAG interface. If the setting has a different value, reset the value to `TX disable` by entering the `loop-protect action tx-disable` command:

```
switch# configure terminal
switch(config)# interface lag 2 multi-chassis
switch(config-if)# loop-protect action tx-disable
```

3. Back up the current software to the secondary image bank. Run the `show version` command. If the `Active Image` is `primary`, copy the primary image to the secondary image bank:

```
switch# copy primary secondary
```

   If the current image bank is secondary, the current 10.00 image is already in the secondary bank.

4. Save a copy of the running configuration on both switches:

```
switch# copy running-config sftp://<username>@<IP>/filename vrf mgmt
```

5. Examine the 10.00 configuration files. Determine the appropriate VSX commands that will need to be run after the software on the device is upgraded. Many MCLAG commands have changed in this release. See **MCLAG to VSX command equivalency** on page 12 for finding the equivalent VSX commands.
6. Check the status of the `show mclag brief` command and validate the ISL is in-sync and the keepalive is established.
7. Check the output of the `show lacp interfaces` command and note which LACP interfaces are in a forwarding state of up.

Procedure

1. Ensure that you are in a maintenance window as both switches will no longer be forwarding network traffic.
2. Copy the software from this release onto both switches by using TFTP, SFTP, or USB:

```
switch# copy sftp://<username>@<ip>/XL_10_01_0001.swi primary vrf mgmt
```

3. Boot each switch to this software release:

```
switch# boot system primary
```

   After each switch has booted, the MCLAG and active gateway configurations are removed.

4. With the information you learned about your configuration in step **5** in prerequisites, configure VSX on each of the switches. Configure one switch with the primary role and its peer switch with the secondary role.

   On the primary switch, enter:

```
switch(config)# vsx
switch(config-vsx)# role primary
switch(config-vsx)# inter-switch-link 1/1/1
switch(config-vsx)# keepalive peer 10.0.0.1 source 10.0.0.2
switch(config-vsx)# exit
switch(config)# interface vlan 10
switch(config-if-vlan)# active-gateway ip 10.1.1.1 mac 00:00:00:00:00:011
```

   On the secondary switch, enter:

```
switch(config)# vsx
switch(config-vsx)# role secondary
switch(config-vsx)# inter-switch-link 1/1/1
switch(config-vsx)# keepalive peer 10.0.0.1 source 10.0.0.2
switch(config-vsx)# exit
switch(config)# interface vlan 10
switch(config-if-vlan)# active-gateway ip 10.1.1.1 mac 00:00:00:00:00:011
```

5. The MCLAG conversion to VSX is complete. Validate on both switches that the ISL is `In-Sync` by running the `show vsx brief` command on both switches. Verify in the output of the command that the keepalive state is `Keepalive-Established`.
6. Validate on both switches that the downstream LACP links are all forwarding correctly by entering the `show lacp interfaces` command.
7. Configure VSX-related features as needed. Consult this guide for additional information about VSX features, such as VSX configuration synchronization and active forwarding.
8. Save the running configuration to the startup configuration:

```
switch# write memory
Success
```

MCLAG to VSX command equivalency

Many of the MCLAG commands have an equivalent VSX command. The following table provides the equivalent commands for MCLAG and VSX. All VSX commands have been moved to a new VSX context, except for active gateway.

Table 1: *Required Commands*

MCLAG	VSX	Description
N/A	`switch(config)# vsx`	VSX contains its own context for configuration. All VSX commands are executed within the VSX context.

`switch(config)# interface 1/1/1`
`switch(config-if)# mclag inter-switch-link`
 	
or
 	
`switch(config)# interface lag1`
`switch(config-lag-if)# mclag inter-switchlink`
	`switch(vsx)# inter-switch-link 1/1/1`
	
	or
	
	`switch(vsx)# inter-switch-link lag 1`
		Configure the interswitch link (ISL) on an interface or a LAG interface.

`switch(config)# mclag device-priority 8`	`switch(vsx)# role primary`
	
	or
	
	`switch(vsx)# role secondary`
		MCLAG uses a device priority value to designate a role. VSX requires the manual configuration of a role. Designate one switch as the primary switch and its peer switch as the secondary switch.

Table Continued

MCLAG	VSX	Description

`switch(config)# keepalive peer 10.0.0.2 source 10.0.0.1`
 	
or
 	
`switch(config)# keepalive peer 10.0.0.2 source 10.0.0.1 red`
	`switch(vsx)# keepalive peer 10.0.0.2 source 10.0.0.1`
	
	or
	
	`switch(vsx)# keepalive peer 10.0.0.2 source 10.0.1 vrf red`
		The keepalive is required and configured under the VSX context.

`switch(config-if)# interface vlan 10`
`switch(config-if-vlan)# active-gateway ip 10.1.1.1 00:00:00:00:00:01`
 	
or
 	
`switch(config-if)# interface vlan 10`
`switch(config-if-vlan )# active-gateway ip6 2001::1 00:00:00:00:00:02`
	`switch(config-if)# interface vlan 10`
	`switch(config-if-vlan)# active-gateway ip 10.1.1.1 mac 00:00:00:00:00:01`
	
	or
	
	`switch(config-if)# interface vlan 10`
	`switch(config-if-vlan )# active-gateway ip6 2001::1 mac 00:00:00:00:00:02`
		If active gateway was used in MCLAG and you want to use active gateway for VSX, configure active gateway again on each interface VLAN.

Table 2: *Optional commands*

MCLAG	VSX	Description
`switch(config)# mclag inter-switch-link hello-interval 2`	`switch(vsx)# inter-switch-link hellointerval`	Configures the ISL hello interval.
`switch(config)# mclag inter-switch-link dead-interval 10`	`switch(vsx)# inter-switch-link deadinterval`	Configures the ISL dead interval.
`switch(config)# mclag inter-switch-link hold-time 1`	`switch(vsx)# inter-switch-link hold-time 1`	Configures the ISL holdtime.
`switch(config)# mclag keepalive udp-port 9999`	`switch(vsx)# keepalive udp-port 9999`	Configures the keepalive UDP port.
`switch(config)# mclag keepalive deadinterval 6`	`switch(vsx)# keepalive dead-interval 6`	Configures the keepalive dead interval.
`switch(config)# mclag keepalive hellointerval 2`	`switch(vsx)# keepalive hello-interval 2`	Configures the keep alive hello interval.

Aruba Virtual Switching Extension (VSX) is virtualization technology for aggregation/core switches running the ArubaOS-CX operating system. This solution lets the switches present as one virtualized switch in critical areas. Configuration synchronization is one aspect of this VSX solution where the primary switch configuration is synced to the secondary switch. This solution allows for a pseudo single pane of glass configuration and helps keep key configuration pieces in synchronization as operational changes are made. Since the solution is primarily for high availability, it is expected that most of the configuration policy is the same across both peers.

VSX virtualizes the control plane of two aggregation switches to function as one device at layer 2 and as independent devices at layer 3. From a datapath perspective, each device does an independent forwarding lookup to decide how to handle traffic. Some of the forwarding databases, such as the MAC forwarding database and neighbor tables, are synchronized between the two devices using a proprietary VSX control plane. Some of the forwarding databases are built independently by each switch.

Benefits of VSX

- Control plane:
  - Dual control plane for better resiliency
  - Unified management (synchronized configuration and easy troubleshooting)
  - Independently software upgradable with near zero downtime
  - In-chassis redundancy for the 8400 series switches and device level redundancy for all other platforms, such as for the 8320 series switches.
- Layer 2 distributed LAGs (aggregation switches to access switches):
  - Loop-free L2 multipathing (active-active)
  - Rapid failover
  - Simple configuration
  - No Spanning Tree Required
- Layer 3 distributed LAGs (core switches to aggregate switches)
  - Distributed Layer 3 over VSX pair (various options: Routed Only Ports (ROPs), Switched Virtual Interfaces (SVIs), or LAG SVIs)
  - Unified datapath (active-active first hop gateway)
  - Layer 3 ECMP and layer 2 VSX (highly fault tolerant)
- Active Gateway:
  - Active-Active first hop gateway (VIP)
  - Simple configuration (one command)

- No gateway protocol overhead
- DHCP relay redundancy

VSX solution topology overview

- **Active gateway support:** Active gateways can be configured for active-active routing. VRRP can be used, as an alternative, for active-standby routing.
- **ISL links assigned to higher bandwidth:** An ISL link has a higher bandwidth compared to VSX links. When planning the topology, consider sizing the ISL link according to the traffic volume required for the east-west traffic of a single-homed VSX during a failover scenario.
- **Reducing traffic loss:** It is recommended, but not required, that each **VSX LAG** has at least two member links. The two links originate from two different line cards on each side of the VSX switches for a total of four member links per switch segment. This configuration reduces the possibility of traffic loss because of a line card reboot. The spanning of member links across line cards is applicable only to 8400 series switches.
- **Same VLAN configurations:** Both VSX switches have the same VLAN configurations. Make sure that no topology loop is formed because an ISL is added as a member to all the VLANs by default. You can make configuration synchronization automatic between the VSX switches by enabling VSX synchronization.
- **Upstream device from VSX switches:** Connections to the upstream device from the VSX switches have sufficient bandwidth to handle traffic from all VSXs.

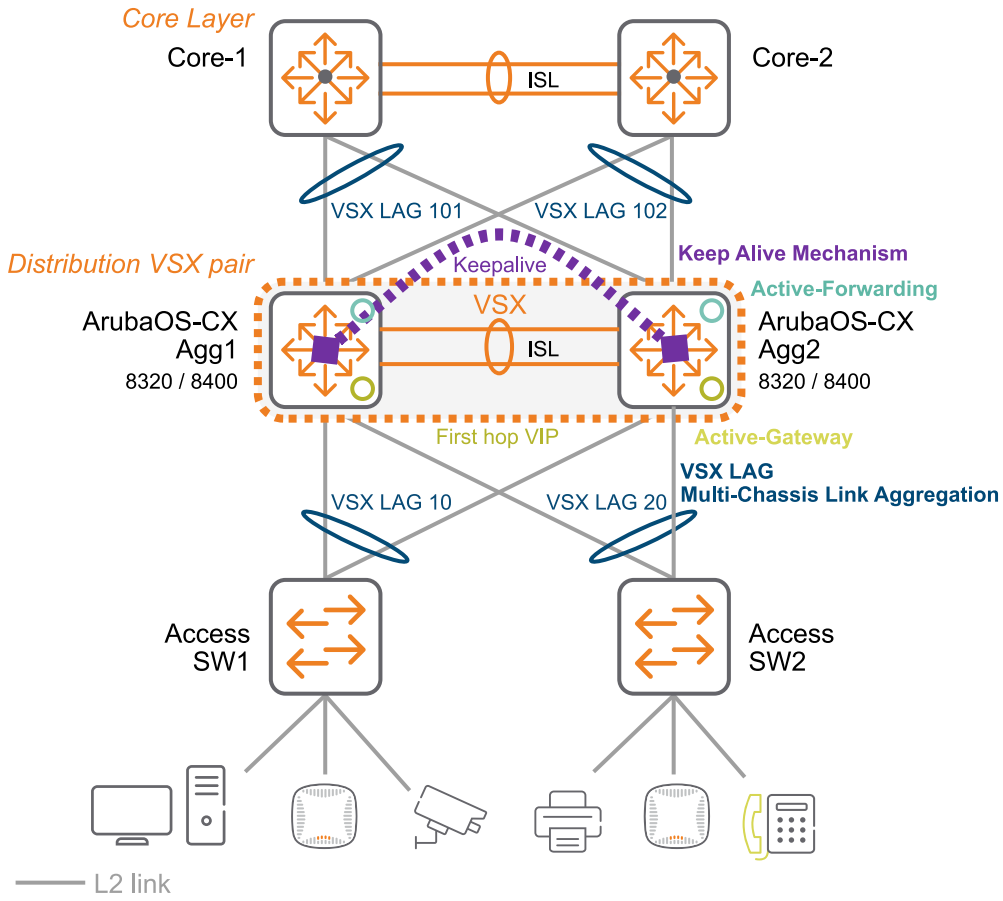
NOTE: Core-1 and Core-2, shown in the following figure, can be third-party devices, as long as they support LACP for downstream connectivity to the VSX LAG. VSX Synchronization syncs from the primary switch (Aggregate-1) to the secondary switch (Aggregate-2).

To configure Core-1 and Core-2 with ArubaOS-CX, see **Configuring core 1 and core 2** on page 28.

To configure the aggregate 1 and aggregate 2, see **Configuring aggregate 1 and aggregate 2 switches** on page 31.

To configure the access switch, see **Configuring the access switch** on page 35.

Figure 1: *Sample VSX solution topology*



More information

Active forwarding on page 26
Benefits of active forwarding and active gateway on page 27
Active gateway over VSX on page 24
Active gateway configurations on page 26

VSX LAG

VSX LAGs span both aggregation switches. The two switches appear as one device to partner downstream or upstream devices or both when forming a LAG with the VSX pair. The two switches synchronize their databases and states over a user configured link referred to as an Inter-Switch Link (ISL).

VSX LAGs are preferable to VLANs because VSX LAGs greatly reduce the number of transit VLANs and associated SVIs, resulting in less latency. You can also greatly decrease the latency by also enabling active forwarding and active gateway. When active forwarding and active gateway is enabled, north-south and south-north traffic bypasses the ISL link.

More information

interface lag multi-chassis on page 43

VSX solution requirements

- **All VSX switches in your environment must have identical settings for the following:**
  - The VLAN membership for all VSX trunk ports.
  - The loop-protection configuration on a VSX VLAN. VSX VLANs are VLANS that have a VSX LAG as a member.
- **IGMP-snooping and VSX VLANs:** IGMP-snooping must be turned off on a VSX VLAN for each VSX switch.
- **Modes for 8320 series switches:** All switches must be either in mode 3 or in mode 4.
- **Spanning tree support:** Spanning tree is not supported with VSX.
- **Support for Inter-Switch links (ISLs):** VSX LAG does not support layer 3 processing, such as a routed port; however, multiple virtual switch interfaces can be configured on the switch in association with the VLANs carried over the given VSX LAG.
- **Support for Layer 3:** A VSX LAG as a route only port is not supported. You can add VSX LAGs to a switch virtual interface to enable Layer 3.
- **VLAN support:** The same VLAN must be supported on both switches because the ISL is automatically made a member of all VLANs created on the switch, except the peer-keepalive VLAN. The VSX solution synchronizes the VLANs on the secondary VSX switches.
- **VSX switches and software versions:** Both VSX peer switches must use the same software version.

More information

Active gateway configurations on page 26
inter-switch-link {<PORT-NUM> | lag <LAG-ID>} on page 44
vsx-sync on page 71

VSX components

Inter-Switch Link (ISL)

In the VSX solution topology, an Inter-Switch Link (ISL) is a layer 2 interface between two VSX peer switches. Each VSX switch must be configured with an ISL link connected to its peer VSX switch. It is recommended that this link is peer-to-peer and used for both datapath traffic forwarding and control path VSX protocol exchange. The ISL interface is by default a member of all VLANs on the device. You can change ISL membership through the command line, but you must ensure VLANs that contain VSX LAG members are not excluded from the ISL.

In the datapath, traffic is forwarded natively with no additional encapsulation, unlike VSF. The ISL link is used to synchronize LACP states, MAC, and ARP tables, in addition to configuration information. The ISL can span long distances (transceiver dependent). Traffic passing over VSX links has no additional encapsulation.

All ISL ports must have the same speed. The speed can be 10G, 40G, or 100G, with 40G and 100G being the preferred speeds. For example: 2x40G.

ISL configurations

Task	Command	Example
Configuring an ISL port.	**inter-switch-link**	`switch(config)# vsx` `switch(config-vsx)# inter-switch-link lag 100`
Deleting an ISL port.	**no inter-switch-link**	`switch(config)# vsx` `switch(config-vsx)# no inter-switch-link`
Configuring ISL dead interval.	**inter-switch-link dead-interval** ***<DEAD-INTERVAL>***	`switch(config)# vsx` `switch(config-vsx)# inter-switch-link dead-interval 10`
Restore default ISL dead interval.	**no inter-switch-link dead-interval**	`switch(config)# vsx` `switch(config-vsx)# no inter-switch-link dead-interval`
Configuring the ISL hello interval.	**inter-switch-link hello-interval** ***<HELLO-INTERVAL>***	`switch(config)# vsx` `switch(config-vsx)# inter-switch-link hello-interval 3`
Restoring default ISL hello interval.	**no inter-switch-link hello-interval**	`switch(config)# vsx` `switch(config-vsx)# no inter-switch-link hello-interval`
Configuring ISL holdtime.	**inter-switch-link hold-time** ***<HOLD-INTERVAL>***	`switch(config)# vsx` `switch(config-vsx)# inter-switch-link hold-time 2`
Restoring default ISL holdtime.	**no inter-switch-link hold-time**	`switch(config)# vsx` `switch(config-vsx)# no inter-switch-link hold-time`

Default values:

- Dead interval: 20 seconds
- Hello interval: 1 second
- Hold time: 0 seconds

Keepalive

Keepalive is a layer 3 interface that is used to exchange heartbeats between VSX peer switches. The heartbeats are exchanged by using the User Datagram Protocol (UDP) and port 7678 (default).

Configure each VSX peer switch with a keepalive connection to the other VSX peer switch. This connection is established over a routed network (IPv4 currently) and does not have to be a dedicated peer-to-peer link unlike ISL. Keepalive packets are UDP-based.

The VSX peer switches should have layer 3 reachability for keepalive interfaces through directly connected interfaces or routed through the upstream layer 3 network. Source of keepalive interfaces can be a layer 3 interface (router port), a loopback interface, or a Switch Virtual Interface (SVI). An SVI is a logical layer 3 interface configured per VLAN (one-to-one mapping) that performs all layer 3 processing for packets to or from all switch ports associated with that VLAN.

NOTE: With respect to the keepalive path, it is highly recommended to separate keepalive traffic from the ISL link.

Use a dedicated layer 3 link.

Keepalive packets can be sourced from the supported layer 3 interface; however, the packet must not be transported over the ISL.

Keepalive response in ISL failure scenarios

- **ISL link is down but the switches are still up and running:** In this case, VSX switches use their keepalive connection to determine that they are both UP and running. Once that is determined, the user configured Primary VSX switch keeps its multichassis (VSX) LAG links up and the secondary VSX switch forces its VSX LAG links to go down with reason. Once the ISL link returns, the MAC and ARP database of the primary switch is synchronized to the secondary switch. Then, the configured delay timer starts. Once the delay timer completes its count, the secondary VSX switch brings up its VSX LAG links.
- **ISL link and one of the VSX switches is down:** The switch that is running sees that the ISL and keepalive connection are both down. Independent of the user configured role (primary or secondary), the switch that is up continues to keep its VSX LAG links up. Subsequently when the peer switch returns, the ISL link comes up first. Then, the returned VSX peer switch synchronizes its MAC and ARP tables from the peer switch that stayed up. After the synchronization completes, the delay time starts. Once the delay timer completes its count, the VSX peer switch brings up its VSX LAG links.

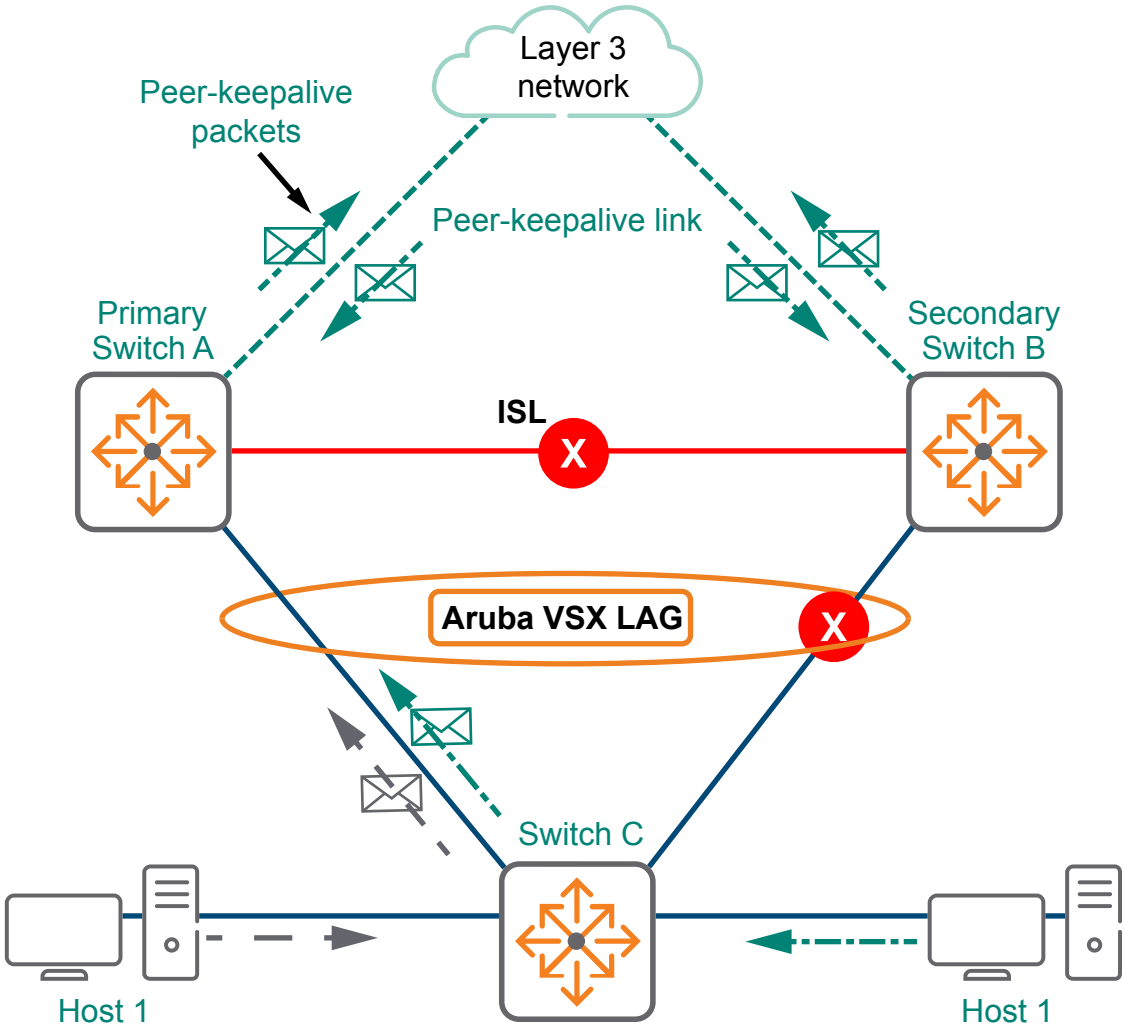
More information

VSX switch reboot on page 36
linkup-delay-timer on page 50
linkup-delay-timer exclude lag-list on page 52

Keepalive scenario

The following diagram illustrates a scenario in which both VSX switches are up, but the ISL link is down. The switches cannot exchange information.

The keepalive functionality brings down the link between Switch B and Switch C in the following diagram. The traffic is forced to go from Switch C to Switch A and then through the Layer 3 network to access Switch B. The keepalive path is over the Layer 3 network. Traffic traveling from Switch B to Switch A is also forced to go through the Layer 3 network.



NOTE: Do not have the keepalive path go over ISL. If the keepalive path uses ISL as its only path and an ISL link failure occurs, the VSX switches would be out of sync without the keepalive functioning.

Keepalive configurations

Task	Command	Example
Configuring keepalive peer source and VRF.	**keepalive peer *<IP-ADDR>* source *<IP-ADDR>* [*<VRF-NAME>*]**	`switch(config-vsx)# keepalive peer 192.168.1.1 source 192.168.1.5 vrf vrf1`
Unconfiguring keepalive.	**no keepalive**	`switch(config-vsx)# no keepalive`
Configuring keepalive UDP port.	**keepalive udp-port *<PORT-NUM>***	`switch(config-vsx)# keepalive udp-port 2000`

Table Continued

Task	Command	Example
Restoring default keepalive UDP port.	**no keepalive udp-port**	`switch(config-vsx)# no keepalive udp-port`
Configuring keepalive hello interval.	**keepalive hello-interval *<HELLO-INTERVAL>***	`switch(config-vsx)# keepalive hello-interval 3`
Restoring default keepalive hello interval.	**no keepalive hello-interval**	`switch(config-vsx)# no keepalive hello-interval`
Configuring keepalive dead interval.	**keepalive dead-interval *<DEAD-INTERVAL>***	`switch(config-vsx)# keepalive dead-interval 10`
Restoring default keepalive dead interval.	**no keepalive dead-interval**	`switch(config-vsx)# no keepalive dead-interval`

Default values:

- Keepalive dead interval: 3 seconds
- Hello interval: 1 second
- UDP port for the keepalive protocol: 7678

Switch roles

Each VSX switch must be configured with a role – primary or secondary. The roles do not indicate which device is forwarding traffic at a given time as VSX is an active-active forwarding solution. The roles are used to determine which device stays active when there is a VSX split, such as when the ISL goes down, and for determining the direction of configuration-sync. If the VSX ISL goes down, the primary switch keeps forwarding traffic while the secondary switch blocks ports from participating in the VSX LAGs.

IGMP snooping

VSX switches can be configured for IGMP snooping on downstream VLANs facing the access switches. When enabled, the IGMP group database is independently constructed on each VSX switch. Multicast traffic to these groups is appropriately pruned/optimized.

Each VSX switch has an identical IGMP group database:

- Each VSX node individually learns any JOIN/LEAVE message received from a downstream VSX LAG.

  For example: Agg-1 learns on downlink from SW1, whereas Agg-2 learns on the ISL as the ISL is always included as a forwarding port for IGMP, as shown in **Figure 1: Sample VSX solution topology** on page 16.

- The VSX IGMP process translates the received IGMP from the ISL into an IGMP join message from the VSX LAG.

Multicast traffic to these IGMP groups is pruned/forwarded based on the individual IGMP group database on each VSX node. ISLP does not synchronize IGMP groups between VSX peers. The IGMP database construction is a data-plane based process.

If a VSX node reboots, it must relearn all the IGMP groups. The VSX switch floods multicast traffic within the VLANs that have active physical ports being forwarded. It then sends an All Hosts Query message. When the VSX node receives all join messages, it relearns and recreates the IGMP groups database.

DHCP relay redundancy

When the two VSX switches are configured for DHCP relay on their VLAN interfaces, only the primary switch actively relays DHCP client requests to the server. The secondary switch acts as a backup. If the primary VSX switch goes down, the secondary switch takes over. Even though both rimary and secondary switches receive the DHCP request, the primary switch takes precedence.

Both devices do not end up relaying DHCP requests to the server as duplicates. That scenario is usually the case with typical aggregation switches running VRRP-based redundancy.

Deployment options for upstream connectivity with active-active forwarding

Aggregate core links can be configured in one of the following ways:

- **Layer 3-LAG/routed ports:** Simple VLAN-free configuration best suited when the network runs on a single VRF domain. With multiple VRFs in the network, one would need multiple routed ports, one per VRF.
- **P2P SVI links:** Each aggregate-core link is on its own VLAN. The layer 2 links can carry traffic for multiple SVIs and therefore multiple VRFs can be carried over the same link.
- **VSX multichassis LAGs:** The aggregate-core links can be multichassis layer 2 links carrying traffic for multiple SVIs and VRFs. This configuration provides for layer 2 LAG and layer 3 ECMP-based active-active forwarding for traffic from core to access.

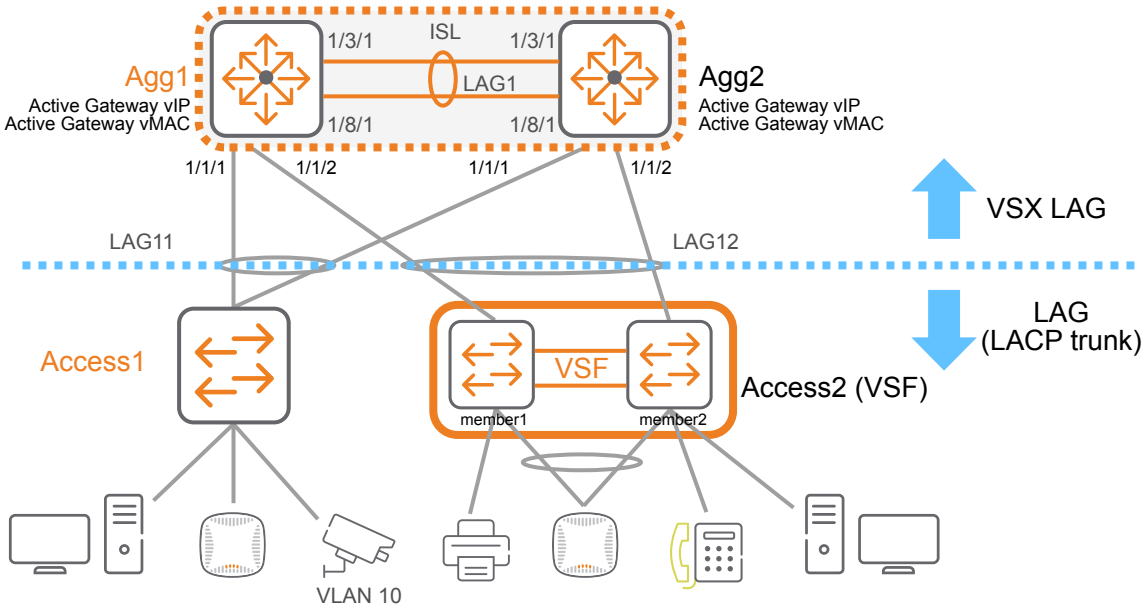
In these configurations, the two VSX switches run as independent control planes (OSPF/BGP) and present themselves as different routers in the network. In the datapath however, they function as a single router and support active-active forwarding.

Active-active layer 2

VSX LAGs span two switches and operate in active-active mode. Traffic between the access layer and aggregation layer switches can be forwarded to any of the active links. There are no loops and no need for spanning tree protocol or blocked ports.

From a datapath perspective, each VSX switch that gets a packet always uses its local links of the LAG to forward traffic to the destination. The VSX switch only uses the ISL link if the local LAG links are down.

Figure 2: *Layer 2 configuration*



The following shows the configuration details from the figure:

```
interface lag 11 multi-chassis
    description access-sw1
    no shutdown
    no routing
    vlan trunk native 1
    vlan trunk allowed 5,10,15,20
    lacp mode active

interface lag 12 multi-chassis
    description access-sw2
    no shutdown
    no routing
    vlan trunk native 1
    vlan trunk allowed 5,10,15,20
    lacp mode active

interface 1/1/1
    no shutdown
    lag 11
interface 1/1/2
    no shutdown
    lag 12
```

Active-active layer 3 default gateway

VSX aggregation switches can be configured with a shared virtual IP/MAC address on the Layer3 VLAN interface that serves as the default gateway for the access layer. The two switches then share the router MAC and function as an active-active gateway for the IP subnet. The first VSX device that receives traffic from the access layer (based on the hashing algorithm over the LAG interface) routes it across to the other subnets. Active gateway is preferable over VRRP because with VRRP traffic is still pushed over the ISL link, resulting in latency in the network.

Key benefits of active gateway compared to VRRP

- **Configuration overhead:** For VRRP, you must configure the VRIDs on per VLAN basis and designate their priorities. Then, you must decide if you have to use an Interface IP address as master IP or configure a new backup IP address.

  With an active gateway, the configuration overhead is simplified. Only the configuration of a virtual IP and a virtual MAC on an interface VLAN is required.

- **Protocol overhead:** With VRRP, the primary VRRP router keeps sending advertisements every second to the backups, which result in a VLAN flood that goes to all nodes in the subnet. The sending of these advertisements helps the backup VRRP router know that the primary VRRP router is still alive.

  With active gateway, there is no protocol advertisement as the configurations are static. Active gateway is active-active so each device does not have to know if the peer device is active from an active gateway perspective.

- **Datapath:** VRRP is an active-standby protocol. If the backup VRRP router receives traffic that must be routed across to the other subnets, the VRRP router bridges it to the primary VRRP router. The primary VRRP router then does the actual forwarding.

  With an active gateway, both devices are ready to forward layer 3 traffic to the other subnets individually. This method avoids the extra bridged hop from the backup to the primary VRRP router.

Active gateway over VSX

Active gateway is a first hop redundancy protocol that eliminates a single point of failure. The active gateway feature is used to increase the availability of the default gateway servicing hosts on the same subnet. An active gateway improves the reliability and performance of the host network by enabling a virtual router to act as the default gateway for that network.

Requirements

- An active gateway can be configured only over an SVI.
- An active gateway can have a maximum 16 "unique" MAC addresses per system, including IPv4 and IPv6 addresses.
- IPv4 and IPv6 active gateways in an SVI cannot share virtual MAC addresses. Given an SVI with both IPv4 and IPv6 active gateways, you must configure a unique active gateway MAC address, so the IPv4 and IPv6 active gateways do not have the same MAC address.
- Only one IPv4 and IPv6 active gateway can be configured on an SVI.
- Link local IPv6 virtual IP address of an active gateway address is multicasted for router advertisement so that the IPv6 address can be chosen as a default gateway.
- Active gateway configuration must be the same in both the VSX peer switches.

Example of IPv4 and IPv6 active gateways on an SVI

Assume that you have IPv4 and IPv6 active gateways on an SVI. Each SVI uses a MAC address for IPv4 and one for IPv6. The configuration of the VSX with an active-gateway consumes a second MAC address per SVI. The following is an example for IPv4:

```
switch# sh int vlan10
```

```
Interface vlan10 is up
Admin state is up
Description: ACCESS switch mgmt
Hardware: Ethernet, MAC Address: 98:f2:b3:68:71:fe
IPv4 address 10.1.1.253/24
Rx
       L3:
            0 packets, 0 bytes
Tx
       L3:
            0 packets, 0 bytes

switch# sh run int vlan141
interface vlan141
   description USER VLAN 10.141.0.0/16
   ip address 10.141.255.253/16
   ip ospf 1 area 0.0.0.0
   ip pim-sparse enable
    ip igmp enable
    ip igmp version 2
   exit
switch# config t
switch(config)# int vlan10
switch(config-if-vlan)# active-gateway ip 10.1.1.254 mac 00:00:00:10:11:12
switch# sh int vlan10

Interface vlan10 is up
Admin state is up
Description: ACCESS switch mgmt
Hardware: Ethernet, MAC Address: 98:f2:b3:68:71:fe
IPv4 address 10.1.1.253/24
active gateway 10.1.1.254          00:00:00:10:11:12
Rx
       L3:
            0 packets, 0 bytes
Tx
       L3:
            0 packets, 0 bytes
```

Example of configuring a virtual IPv4 and IPv6 address for an interface VLAN

Configuring a virtual IPv4 and IPv6 address for an interface VLAN:

```
switch# config
switch(config)# vlan 2
switch(config)# interface vlan 2
switch(config-if-vlan)# ip address 10.0.0.1/24
switch(config-if-vlan)# active-gateway ip 10.0.0.2 mac 00-00-5E-00-53-FF
switch(config-if-vlan)# ipv6 address aa:bb::cc:dd/24
switch(config-if-vlan)# active-gateway ipv6 2001:DB8::/32 mac 00-00-5E-00-53-00
```

Viewing the configuration from the previous example:

```
switch(config-if-vlan)# show running-config interface vlan2
interface vlan2
   ip address 10.0.0.1/24
   active-gateway ip 10.0.0.2 mac 00-00-5E-00-53-FF
   active-gateway ipv6 2001:DB8::/32 mac 00-00-5E-00-53-03
   ipv6 address aa:bb::cc:dd/24
   exit
```

Active gateway configurations

Task	Command	Example
Configuring active gateway for active-active routing.	**active-gateway {ip *<IPV4-ADDR>* | ipv6 *<IPV6-ADDR>*} [*<VMAC-ADDR>*]**	`switch(config-if-vlan)# active-gateway ip 192.168.1.253 01:01:01:01:01:02` `switch(config-if-vlan)# active-gateway ipv6 fe80::00 01:01:01:01:01:02`
Unconfiguring active gateway for active-active routing.	**no active-gateway {ip | ipv6}**	`switch(config-if-vlan)# no active-gateway ip` `switch(config-if-vlan)# no active-gateway ipv6`

VRRP with VSX configuration

VRRP is similar to active gateway in that it is a first hop redundancy protocol that eliminates a single point of failure. One VSX switch acts as a VRRP master and the other switch acts as the VRRP backup. Both VSX switches route the traffic. The active gateway/VRRP configuration must be consistent across the two VSX switches. The active gateway/VRRP are mutually exclusive configurations at the switch.

NOTE: Active gateway is preferable to VRRP because VRRP traffic is still pushed over the ISL link, resulting in latency.

Sample VRRP configuration

```
switch# configure
switch(config)# vlan 1-10
switch(config)# router vrrp enable
switch(config)# interface vlan2
switch(config-if-vlan)# ip address 192.168.1.253/16
switch(config-if-vlan)# no shutdown
switch(config-if-vlan)# vrrp 1 address-family ipv4
switch(config-if-vrrp)# address 192.168.1.253 primary
switch(config-if-vrrp)# no shutdown
switch(config-if-vrrp)# exit
switch(config-if-vlan)# exit
switch(config)#
```

Active forwarding

Active forwarding is an optimization for layer 3 unicast traffic flowing from the upstream (core) to the downstream (access) through the VSX peers (aggregate). Active forwarding prevents the bridged traffic from switching over the ISL. It also minimizes latency and the ISL bandwidth.

Active forwarding requirements

- Active forwarding is enabled on a SVI facing core network on a VSX environment.
- Active forwarding is supported on SVI only.
- Active forwarding and active gateway are mutually exclusive on the same SVI.
- A maximum of 16 active forwarding SVIs are supported per switch.

Traffic flow scenario

Active forwarding mitigates the suboptimal path scenarios because of undeterministic layer 3 hashing and layer 2 hashing, as described in the following ECMP (equal-cost multi-path routing) scenario.

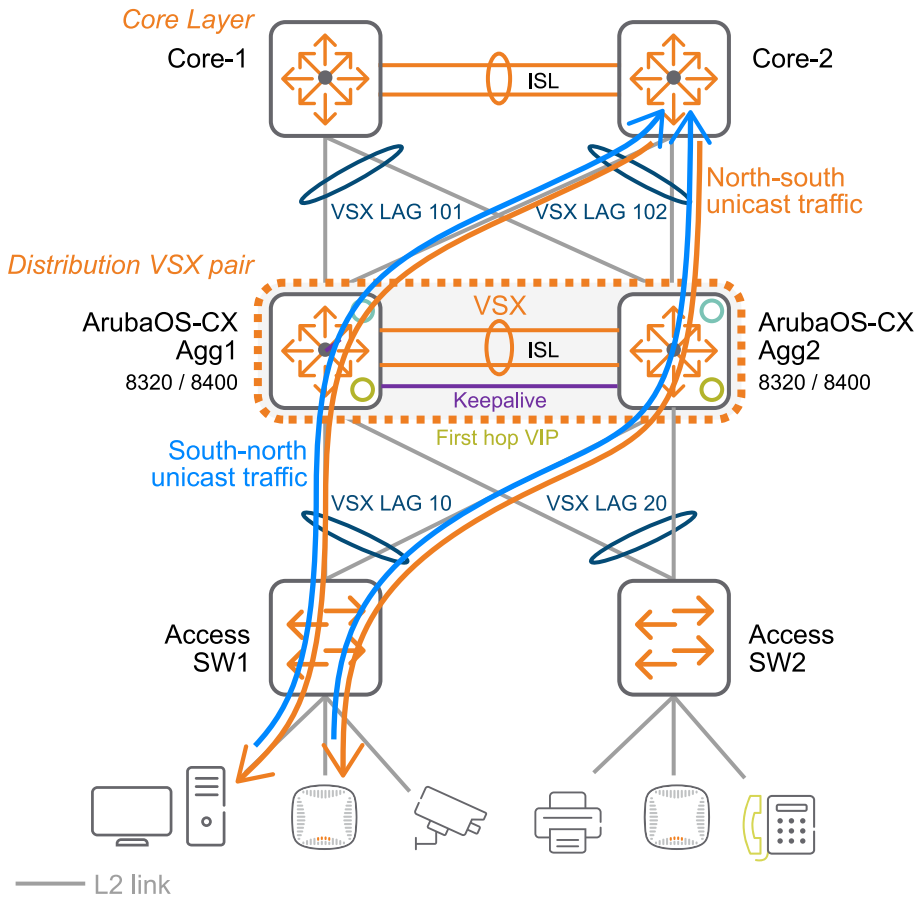
This scenario describes a situation when active forwarding is not used. In a VSX environment, a core network is connected to a VSX pair, forming an OSPF adjacency over a VSX LAG. The VSX LAG has ECMP routes to the access network. The core has ECMP routes to choose between either the VSX primary switch or the VSX secondary switch for traffic flowing from the core to the access network. Assume that ECMP picked the VSX primary switch. This traffic is now subjected to the hashing algorithm over the VSX LAG interface. Based on the chosen hashing algorithm, the layer 2 interface might route the traffic to the VSX secondary switch. The secondary VSX switch then bridges this traffic over the ISL to the primary VSX switch. The primary VSX switch in turn routes the traffic toward the access network, which causes extra overhead with ISL bandwidth and network latency.

If active forwarding was enabled in the previous scenario, the traffic destined for the access network would not be bridged over the ISL, instead the traffic would flow from north to south, resulting in less network latency. For more information about the benefits of active forwarding, along with a diagram, see **Benefits of active forwarding and active gateway** on page 27.

Benefits of active forwarding and active gateway

The enabling of active forwarding and active gateway reduces latency in the network by bypassing the ISL link for north-south and south-north traffic, resulting in one less hop.

When active forwarding is enabled, the north-south unicast traffic bypasses the ISL link for Agg1 and Agg2. Just as the south-north traffic bypasses the ISL link for Agg1 and Agg2 when active gateway is enabled, as shown in the following figure.



Configuring core 1 and core 2

The steps in this section are for configuring core 1 and core 2 as shown in **Figure 1: Sample VSX solution topology** on page 16.

After completing these steps, configure the aggregate 1 and aggregate 2 switches in your network topology, as described in **Configuring aggregate 1 and aggregate 2 switches** on page 31. Then, enable VSX configuration synchronization for a feature, as described in **Enabling VSX configuration synchronization for a feature** on page 34.

A VSX LAG supports a maximum of four member links per switch segment. A VSX LAG across a downstream switch can have at most a total of eight member links. Run the `show capacities` command for the maximum number of VSX LAGs supported for your type of switch.

The core can be third-party devices, as long as they support LACP for downstream connectivity to the VSX LAG. VSX synchronization syncs from the primary switch (aggregate 1) to the secondary switch (aggregate-2).

NOTE: When creating a VSX LAG, select an equal number of member links in each segment for load balancing, such as four member links (one segment) and four member links (another segment). Do not create a VSX LAG with six member links in one switch and two member links on another segment.

Procedure

1. Access the prompt on the switch you want to make the primary core switch:

```
switch# config
```

2. If the switch does not have a hostname, create one:

```
switch(config)# hostname <HOSTNAME>
```

3. Create the required VLANS:

```
switch(config)# vlan 1-20
```

4. Enable OSPFv2:

```
switch(config)# router ospf 1
switch(config-ospf-1)# redistribute connected
switch(config-ospf-1)# area 0.0.0.0
```

5. Enable OSPFv3:

```
switch(config)# router ospfv3 1
switch(config-ospfv3-1)# redistribute connected
switch(config-ospfv3-1)# area 0.0.0.0
switch(config-ospfv3-1)# exit
```

6. Create a loop back interface and enable OSPFv2/v3:

```
switch(config)# interface loopback 1
switch(config-loopback-if)# ip address 3.3.3.3/24
switch(config-loopback-if)# ip ospf 1 area 0.0.0.0
switch(config-loopback-if)# ipv6 ospfv3 1 area 0.0.0.0
switch(config-loopback-if)# exit
```

7. Enable OSPFv2/v3 on the physical port:

```
switch(config)# interface 1/2/43
switch(config-if)# no shutdown
```

```
switch(config-if)# ip address 192.168.10.5/24
switch(config-if)# ipv6 address 2001:11::3/64
switch(config-if)# ip ospf 1 area 0.0.0.0
switch(config-if)# ipv6 ospfv3 1 area 0.0.0.0
switch(config-if)# exit
```

8. Create a VLAN for the host network:

```
switch(config)# interface vlan 200
switch(config-if-vlan)# ip address 192.168.10.6/16
switch(config-if-vlan)# ipv6 address 2001:200::1/64
switch(config-if-vlan)# exit
```

9. Enable the port for host communication:

```
switch(config)# interface 1/1/48
switch(config-if)# no shutdown
switch(config-if)# no routing
switch(config-if)# vlan access 200
switch(config-if)# exit
```

10. Enter `vsx`:

```
switch(config)# vsx
switch(config-vsx)#
```

11. Enter the **role primary** command for assigning the primary role to a switch. If you have already gone through these steps for configuring the primary switch and you are now configuring the secondary switch, enter the **role secondary** command

    Setting the primary role on a switch:

```
switch(config-vsx)# role primary
```

    Setting the secondary role on a switch:

```
switch(config-vsx)# role secondary
```

12. Configure a layer 2 interface as an ISL:

```
switch(config-vsx)# inter-switch-link lag 100
```

    In this instance, an ISL was configured over LAG 100.

NOTE: Before you enter this command, verify that the interface is layer 2 and the LAG is not a VSX LAG.

13. The steps in this topic are based on a topology (**Figure 1: Sample VSX solution topology** on page 16) that shows a keepalive path over the core. To configure the keepalive interface as a loopback interface:

```
switch(config)# int loopback 0
switch(config-loopback-if)# ip address 192.168.1.1/32
switch(config-loopback-if)# ip ospf 1 area 0
switch(config-loopback-if)# exit
switch(config)# vsx
switch(config-vsx)# keepalive peer 192.168.1.2 source 192.168.1.1
```

NOTE: The source of the keepalive interface can be a supported layer 3 interface through the loopback interface, SVI, or layer 3 interface. The source must be reachable to the VSX peer through layer 3. The path can be over the core or direct path. The keepalive path must not be over the ISL.

14. Change the context to the `switch(config)#` context:

```
switch(config-vsx)# exit
switch(config)#
```

15. Configuring a LAG interface as an ISL:

```
switch(config)# interface lag <LAG-ID>
```

For example, configuring LAG 102 as an ISL LAG:

```
switch(config)# interface lag 102
switch(config)# vsx
switch(config)# inter-switch-link lag 102
```

16. Repeat the previous steps for the secondary core switch.
17. Enter the **show vsx configuration inter-switch-link** command for confirming the properties of the VSX LAG, such as confirming if the ISL is in-sync.

```
switch# show vsx configuration inter-switch-link
Inter Switch Link    : 1/1/1
Hello Interval       : 1 Seconds
Dead Interval        : 20 Seconds
Hold Time            : 0 Seconds
Device Role          : primary
Multichassis LAGs    : lag 101
```

More information

keepalive peer on page 49
interface lag multi-chassis on page 43

VSX configuration synchronization

VSX configuration synchronization simplifies VSX solution management, reduces configuration misconfiguration, and drift across VSX peer switches. With configuration synchronization enabled, the primary peer configuration is synced to the secondary peer. This synchronization is controlled in an opt-in manner by enabling VSX synchronization on a section of configuration.

If one or more of the following scenarios occur, the secondary switch will receive the configuration update after it fulfills synchronization requirements and is fully enabled:

- The secondary switch is not present.
- The secondary switch is not connected to the primary switch through the ISL.
- The secondary switch is not configured for VSX configuration synchronization at the time VSX configuration synchronization is enabled on the primary switch.

You can only enable a specific configuration for syncing through the `vsx-sync` CLI extension on the primary switch. This extension is blocked on the secondary peer switch except when VSX configuration-synchronization is disabled or the ISL link is down.

Attributes supporting VSX

You can enable VSX synchronization for:

- Access lists
- Access lists and VLANs associated with a system or LAG interface.
- Active gateways associated with a VLAN interface
- Classes
- Policies
- VLANs

VSX synchronization requirements

- Software image versions must be the same on both switches.
- Primary and secondary roles configured.*
- An interswitch link must be configured.*
- Keepalive*

*Steps on to how meet these requirements are provided in **Configuring aggregate 1 and aggregate 2 switches** on page 31.

Configuring aggregate 1 and aggregate 2 switches

The steps in this section are for configuring the aggregate 1 and aggregate 2 VSX switches, as described in **Figure 1: Sample VSX solution topology** on page 16. VSX switches do not automatically have VSX configuration synchronization enabled. After completing the steps in this section, enable VSX configuration synchronization for a feature, as described in **VSX configuration synchronization** on page 30. VSX synchronization sync configuration information from the primary switch (Aggregate-1) to the secondary switch (Aggregate-2).

VSX switches do not automatically have VSX configuration synchronization enabled. After completing the steps in this section, enable VSX configuration synchronization for a feature, as described in **VSX configuration synchronization** on page 30.

A VSX LAG supports a maximum of four member links per switch segment. A VSX LAG across a downstream switch can have at most a total of eight member links. Run the `show capacities` command for the maximum number of VSX LAGs supported for your type of switch.

NOTE: When creating a VSX LAG, select an equal number of member links in each segment for load balancing, such as four member links (one segment) and four member links (another segment). Do not create a VSX LAG with six member links in one switch and two member links on another segment.

Procedure

1. Access the prompt on the switch you want to make the primary aggregate switch:

   ```
   switch# config
   ```

2. If the switch does not have a hostname, create one:

   ```
   switch(config)# hostname <HOSTNAME>
   ```

3. Create the required VLANS:

```
switch(config)# vlan 1-20
```

4. Create the ISL interface:

```
switch(config)# interface lag 128
switch(config-lag-if)# no shutdown
switch(config-lag-if)# no routing
switch(config-lag-if)# vlan trunk native 1
switch(config-lag-if)# lacp mode active
```

   In this example, LAG 128 is being used as the ISL.

5. Add a physical interface into the LAG:

```
switch(config)# interface 1/4/28
switch(config-if)# no shutdown
switch(config-if)# lag 128
switch(config)# interface 1/4/32
switch(config-if)# no shutdown
switch(config-if)# lag 128
```

6. Enable the interface for keepalive communication:

```
switch(config)# interface 1/1/5
switch(config-if)# ip address 192.168.100.1/24
switch(config-if)# ipv6 address 2001:13::1/64
```

7. Go to the `vsx` context:

```
switch(config)# vsx
switch(config-vsx)#
```

8. Enter the **role primary** command for assigning the primary role to a switch. If you have already gone through these steps for configuring the primary switch and you are now configuring the secondary switch, enter the **role secondary** command

   Setting the primary role on a switch:

```
switch(config-vsx)# role primary
```

   Setting the secondary role on a switch:

```
switch(config-vsx)# role secondary
```

9. Enable ISL:

```
switch(config-vsx)# inter-switch-link lag 128
```

   In this example, ISL is being enabled for LAG 128.

NOTE: Before you enter this command, verify that the interface is layer 2 and the LAG is not a VSX LAG.

10. Enable keepalive:

```
switch(config-vsx)# keepalive peer 192.168.100.2 source 192.168.100.1
```

   In this example, 192.168.100.2 is the peer IP address and 192.168.100.1 is the source IP address.

11. Enable the multichassis interface:

```
switch(config)# interface lag 1 multi-chassis
switch(config-lag-if)# no shutdown
```

```
switch(config-lag-if)# no routing
switch(config-lag-if)# vlan trunk native 1
switch(config-lag-if)# vlan trunk allowed 11
```

12. Add physical interfaces into the multichassis interface:

```
switch(config)# interface 1/1/1
switch(config-if)# no shutdown
switch(config-if)# lag 1
```

13. Create an active gateway SVI:

```
switch(config)# interface vlan 11
switch(config-if-vlan)# ip address 192.168.100.5/16
switch(config-if-vlan)# ipv6 address 2001:DB8::2/64
switch(config-if-vlan)# active-gateway ip 192.168.100.5 mac 00:00:00:00:00:01
switch(config-if-vlan)# active-gateway ipv6 2001:DB8::2 mac 00:00:01:00:00:01
```

14. Enable uplink communication for OSPFv2:

```
switch(config)# router ospf 1
switch(config-ospf-1)# redistribute connected
switch(config-ospf-1)# area 0.0.0.0
```

15. Enable uplink communication for OSPFv3:

```
switch(config)# router ospfv3 1
switch(config-ospfv3-1)# redistribute connected
switch(config-ospfv3-1)# area 0.0.0.0
```

16. Create the loopback interface and enable OSPFv2/v3:

```
switch(config)# interface loopback 1
switch(config-loopback-if)# ip address 192.168.100.6/24
switch(config-loopback-if)# ip ospf 1 area 0.0.0.0
switch(config-loopback-if)# ipv6 ospfv3 1 area 0.0.0.0
```

17. Enable OSPFv2/v3 on the physical port:

```
switch(config)# interface 1/4/30
switch(config-if)# no shutdown
switch(config-if)# ip address 192.168.100.6/24
switch(config-if)# ipv6 address 2001:11::1/64
switch(config-if)# ip ospf 1 area 0.0.0.0
switch(config-if)# ipv6 ospfv3 1 area 0.0.0.0
```

18. Enable OSPFv2/v3 on the physical interface:

```
switch(config)# interface 1/1/5
switch(config-if)# ip ospf 1 area 0.0.0.0
switch(config-if)# ipv6 ospfv3 1 area 0.0.0.0
```

19. Repeat the previous steps for the secondary aggregate switch.

20. View the running configuration by entering the following on the primary and secondary switches:

```
switch# show running-config
```

21. Verify that the ISL link is In-Sync, the role of the switch, and the keepalive state (if enabled) by entering the following on the primary and secondary switches:

```
switch# show vsx brief
```

22. Verify the VSX status by entering the following on the primary and secondary switches:

```
switch# show vsx status
```

23. Verify the LACP interface status by entering the following on the primary and secondary switches:

```
switch# show lacp interfaces
```

24. Verify the uplink (layer 3 communication) by entering the following on the primary and secondary switches:

```
switch# show ip ospf neighbors
```

More information

Benefits of active forwarding and active gateway on page 27
inter-switch-link {<PORT-NUM> | lag <LAG-ID>} on page 44
role {primary | secondary} on page 53
interface lag multi-chassis on page 43
keepalive peer on page 49

Enabling VSX configuration synchronization for a feature

Feature	Command for enabling	Example
Access lists associated with interface	**vsx-sync {[vlans access-lists] | [access-lists]}**	`switch(config)# interface 1/1/1` `switch(config-if)# vsx-sync access-lists`

Access lists associated with a LAG	**vsx-sync {[vlans access-lists] | [access-lists]}**	`switch(config)# interface lag 2`
		`switch(config-lag-if)# vsx-sync access-lists`
		`switch(config-lag-if)# apply access-list ip ITBoston in`

One or more active gateways associated with the interface	**vsx-sync active-gateways**	`switch(config)# interface vlan 1`
		`switch(config-if-vlan)# vsx-sync active-gateways`
		`switch(config-if-vlan)# active-gateway ip 10.10.10.10 23:24:25:26:27:28`
		`switch(config-if-vlan)# active-gateway ip6 fd12:3456:789a:1::1 23:24:25:26:27:28`

Class context to the secondary peer	**vsx-sync**	`switch(config)# class ip ITHouston` `switch(config-class-ip)# vsx-sync`
Entire access list to the secondary peer	**vsx-sync**	`switch(config)# access-list ip ITBoston` `switch(config-acl-ip)# vsx-sync`
Entire policy context to the secondary peer	**vsx-sync**	`switch(config)# policy ITPaloAlto` `switch(config-policy)# vsx-sync`
VLAN context to the secondary peer	**vsx-sync**	`switch(config)# vlan 1` `switch(config-vlan-1)# vsx-sync`

Table Continued

Feature	Command for enabling	Example

VLANs associated with a LAG	**vsx-sync {[vlans access-lists] | [vlans]}**	`switch(config)# interface lag 1`
		`switch(config-lag-if)# vsx-sync vlans`
		`switch(config-lag-if)# vlan trunk native 1`

VLANs associated with interface	**vsx-sync {[vlans access-lists] | [vlans]}**	`switch(config)# interface 1/1/1` `switch(config-if)# vsx-sync vlans`
VLANs and access lists associated with interface	**vsx-sync {[vlans access-lists] | [vlans]}**	`switch(config)# interface 1/1/1` `switch(config-if)# vsx-sync vlans access-lists`

VLANs and access lists associated with a LAG	**vsx-sync {[vlans access-lists] | [vlans]}**	`switch(config)# interface lag 3`
		`switch(config-lag-if)#vsx-sync vlans access-lists`
		`switch(config-lag-if)# vlan trunk native 1`
		`switch(config-lag-if)# apply access-list ip ITPaloAlto in`

VSX active-forwarding on an interface VLAN	**vsx active-forwarding**	`switch# interface vlan 3`
		`switch(config-if-vlan)# vsx active-forwarding`
		`switch(config-vsx)#`

Configuring the access switch

Procedure

1. Access the prompt on the access switch:

```
switch# config
```

2. If the switch does not have a hostname, create one:

```
switch(config)# hostname <HOSTNAME>
```

3. Create an LACP:

```
switch(config)# trunk a3,a4 trk1 lacp
```

4. Create a VLAN:

```
switch(config)# vlan 11
```

5. If you plan to have multiple access switches in your topology, repeat the previous steps.
6. Verify the running configuration:

```
switch# show running-config
```

Consistency checking between VSX switches

Use the following commands to verify that all configurations are in-sync between VSX switches. These commands are helpful in troubleshooting configuration mismatches across VSX peer switches.

Task	Command
Displaying the VSX global configuration consistency between two VSX switches. Use this command to troubleshoot configuration mismatches across VSX peer switches.	**show vsx config-consistency**
Displaying VSX LACP configuration consistency between two VSX switches. Use this command to troubleshoot configuration mismatches across VSX peer switches.	**show vsx config-consistency lacp [<*LAG-NAME*>]**

Viewing the show commands for both VSX switches from one switch

You can view the outputs of the show command for the primary and secondary VSX switches from one switch. When you enter a show command with the `vsx-peer` parameter, the command displays the output from the peer device.

For example, the following command was entered on the primary switch. The `vsx-peer` parameter indicates to the software to display the output as if the command was entered on the secondary switch.

```
switch# show vsx status vsx-peer
VSX Operational State
---------------------
  ISL channel             : In-Sync
  ISL mgmt channel        : operational
  Config Sync Status      : in-sync
  NAE                     : peer_reachable
  HTTPS Server            : peer_reachable

Attribute           Local               Peer
------------        --------            --------
ISL link            lag1                lag1
ISL version         2                   2
System MAC          e0:07:1b:cb:72:e4   98:f2:b3:68:79:2e
Platform            8320                8320
Software Version    TL.10.01.0002B      TL.10.01.0002B
Device Role         secondary           primary
```

NOTE: The show commands that display the file system contents, such as `show logging` or `show core-dump`, do not support the `vsx-peer` parameter.

If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed.

VSX switch reboot

After a VSX switch reboots, it has no entries for ARP, MAC, and routes. If downstream VSX LAG ports are activated before all this information is relearned, traffic is dropped. To avoid a traffic drop, VSX LAGs on the rebooted switch stay down until the restoration of LACP, MAC, and ARP databases.

The learning process for the VSX LAGs has two phases:

- **Initial sync phase:** The LACP states, MAC address table, and ARP table are downloaded from the forwarding switch to the freshly rebooted switch.
- **Link-up delay phase:** The downloaded entries are installed into the ASIC. Router adjacencies with core nodes and learned upstream routes are also established.

  The link-up delay phase is configurable with the `linkup-delay-timer <DELAY-TIMER>` command. The default value is 180 seconds. Set the link-up delay timer to the maximum value of 600 seconds for a network with many MAC addresses, a large ARP table, or a large routing table.

When both VSX switches reboot, the link-up delay timer is not used because both switches are trying to relearn the LACP states, MAC address table, and ARP table.

To get upstream router adjacencies established during the link-up delay, the upstream LAGs have to be excluded from the scope of the link-up delay. Run the `linkup-delay-timer exclude lag-list <LAG-LIST>` for identifying the LAGs for exclusion.

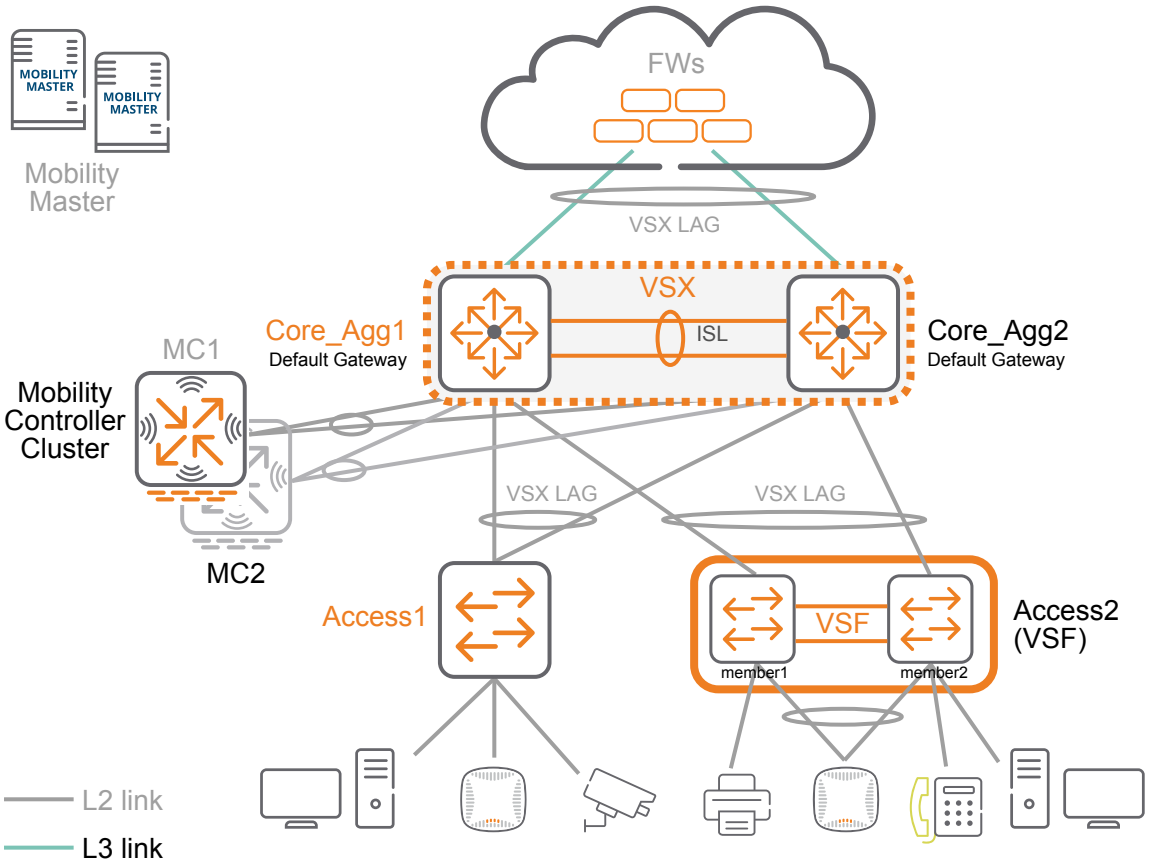
For example, assume that you have a topology similar to the one in **Figure 1: Sample VSX solution topology** on page 16, the upstream LAGs (LAG 101 and LAG 102), would need to be identified by the `linkup-delay-timer exclude lag-list <LAG-LIST>` for exclusion before a VSX switch reboot.

More information

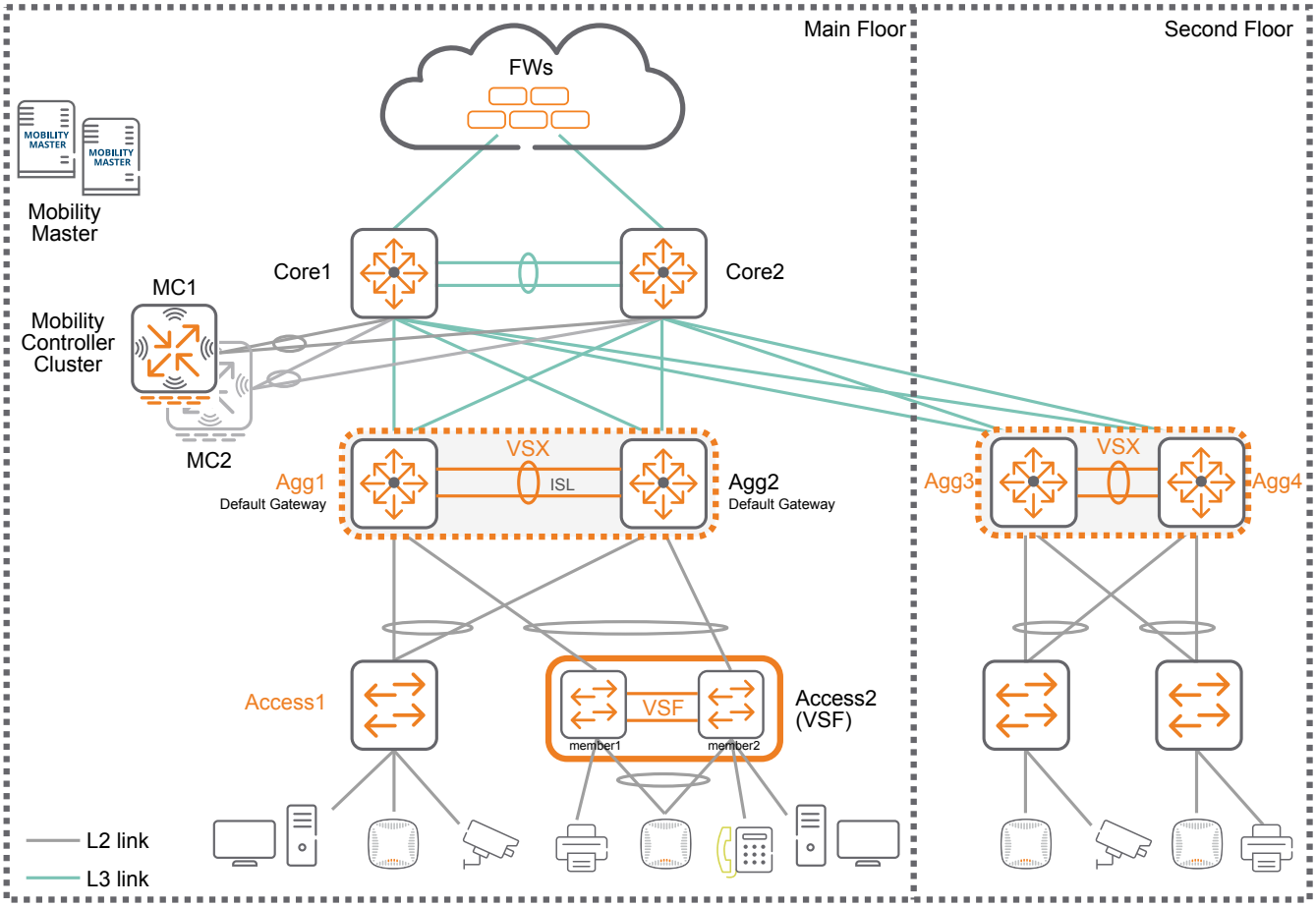
linkup-delay-timer on page 50
linkup-delay-timer exclude lag-list on page 52

Sample campus topologies

Two-tier campus network (small campus-VSX)



Three-tier campus network - medium size (traditional topology)



Troubleshooting

ISL is out-of-sync

Solution 1

Cause

Mismatch with the ISL version or switch platform or both.

Action

1. Run the `show vsx status` command.

   In the following example, the ISL channel is shown as in-sync; however, if the ISL channel was not in-sync, a different status would be provided.

```
switch# show vsx status
VSX Operational State
---------------------
  ISL channel             : In-Sync
  ISL mgmt channel        : operational
  Config Sync Status      : in-sync
  NAE                     : peer_unreachable
```

```
  HTTPS Server            : peer_unreachable

Attribute           Local                       Peer
------------        --------                    --------
ISL link            1/1/43                      1/1/43
ISL version         0                           0
System MAC          48:0f:cf:af:70:84           48:0f:cf:af:c2:84
Platform            8320                        8320
Software Version    TL.10.01.00004              TL.10.01.00004
Device Role         primary                     secondary
```

2. If there is an ISL version mismatch, update the software so the ISL version is the same on the local and peer VSX switch.
3. If the switches have mismatching platforms, create an ISL link that connects two VSX switches with the same platform.

Solution 2

Cause

The role is not configured on any of the VSX switches or the same role is configured on both VSX switches.

Action

1. Run the `show vsx status` command.

   In the following example, the roles are configured correctly; however, if the device roles were configured incorrectly, you might see two primary or two secondary roles or a missing role in the following output.

```
switch# show vsx status
VSX Operational State
---------------------
  ISL channel             : In-Sync
  ISL mgmt channel        : operational
  Config Sync Status      : in-sync
  NAE                     : peer_unreachable
  HTTPS Server            : peer_unreachable

Attribute           Local                       Peer
------------        --------                    --------
ISL link            1/1/43                      1/1/43
ISL version         0                           0
System MAC          48:0f:cf:af:70:84           48:0f:cf:af:c2:84
Platform            8320                        8320
Software Version    TL.10.01.00004              TL.10.01.00004
Device Role         primary                     secondary
```

2. Set the roles correctly so that one of the VSX switches has the primary role and the other switch has the secondary role. To set a switch role, enter the `role {primary | secondary}` command.

Solution 3

Cause

The ISL interface is down on any one switch in the VSX pair.

Action

1. Check ISL state and ISL link status by entering: `switch# show vsx status inter-switch-link`

In the following example, the ISL state and link status are shown as in-sync and up; however, if the ISL interface is down, a different status would be provided.

```
switch# show vsx status inter-switch-link
State                     : In-Sync
Link Status               : up
Mgmt state                : operational

Inter-switch link Statistics
----------------------------
Hello Packets Tx          : 4572
Hello Packets Rx          : 4573
Data Packets Tx           : 80634
Data Packets Rx           : 80637
Mgmt Packets Tx           : 25946
Mgmt Packets Rx           : 25167
Mgmt Packet Drops         : 0
```

2. Re-enable the ISL interface by going to that interface context and entering `no shutdown`:

```
switch(config)# interface 1/1/1
switch(config-if)# no shutdown
```

Traffic drop on a VSX LAG interface

Action

1. Verify that the VSX LAG interface is in-sync with both peer and down stream switches by entering:
   `switch# show lacp interfaces multi-chassis`
2. Verify that the VLAN membership of the VSX is the same on both VSX switches by entering:
   `switch# show vsx config-consistency lacp <LAG-NAME>`
3. Verify that all MAC addresses are programmed correctly on both VSX switches by entering:
   `switch# show mac-address-table`
   `switch# show mac-address-table count`

VSX commands

active-gateway

Syntax

```
active-gateway {ip | ipv6} <IP-ADDRESS> mac <VMAC-ADDRESS>

no active-gateway {ip | ipv6}
```

Description

Configures a virtual IP and virtual MAC for an interface VLAN. A maximum of 256 active-gateways (IPv4 and IPv6) can be configured on the 8320 series switches, and a maximum of 512 active-gateways (IPv4 and IPv6) can be configured on the 8400 series switches.

The `no` form of this command removes the active gateway for active-active routing.

NOTE: This `active-gateway {ip | ipv6} <IP-ADDRESS> mac <VMAC-ADDRESS>` command replaces the `active-gateway {ip <IPV4-ADDR> | ipv6 <IPV6-ADDR>} [<VMAC-ADDR>]` command that is supported only in 10.00 releases.

Command context

`config-if-vlan`

Parameters

`ip`

Specifies the configuration of an IPv4 address.

`ipv6`

Specifies the configuration of an IPv6 address.

`<IP-ADDRESS>`

Specifies the IPv4 or IPv6 address.

- Syntax for IPv4: `A.B.C.D`
- Syntax for IPv6: `A:B::C:D`

`<VMAC-ADDR>`

Specifies the Virtual MAC address. Syntax: `xx:xx:xx:xx:xx:xx`

Authority

Administrators

Usage

Active forwarding cannot be configured when ICMP redirect is enabled. Enter the `no ip icmp redirect` command for disabling ICMP redirect.

Examples

Configuring a virtual IPv4 and IPv6 address for an interface VLAN:

```
switch# config
switch(config)# vlan 2
switch(config)# interface vlan 2
switch(config-if-vlan)# ip address 10.0.0.1/24
switch(config-if-vlan)# active-gateway ip 10.0.0.2 mac 00-00-5E-00-53-FF
switch(config-if-vlan)# ipv6 address aa:bb::cc:dd/24
switch(config-if-vlan)# active-gateway ipv6 2001:DB8::/32 mac 00-00-5E-00-53-00
```

Viewing the configuration from the previous example:

```
switch(config-if-vlan)# show running-config interface vlan2
interface vlan2
   ip address 10.0.0.1/24
   active-gateway ip 10.0.0.2 mac 00-00-5E-00-53-FF
   active-gateway ipv6 2001:DB8::/32 mac 00-00-5E-00-53-03
   ipv6 address aa:bb::cc:dd/24
   exit
```

Removing the active gateway for active-active routing:

```
switch(config-if-vlan)# no active-gateway ip
switch(config-if-vlan)# no active-gateway ipv6
```

config-sync disable

Syntax

```
config-sync disable

no config-sync disable
```

Description

Pauses the VSX synchronization from the primary VSX device to the secondary VSX device.

The `no` form of this command restarts the VSX synchronization.

Command context

```
config-vsx
```

Authority

Administrators

Example

Pauses the VSX configuration synchronization:

```
switch# vsx
switch(config-vsx)# config-sync disable
```

Enables the VSX configuration synchronization:

```
switch# vsx
switch(config-vsx)# no config-sync disable
```

interface lag multi-chassis

Syntax

```
interface lag <LAG-ID> multi-chassis

no interface lag <LAG-ID>
```

Description

Configures a given LAG as a multichassis LAG (VSX LAG), which supports a maximum of four member links per switch segment. A VSX LAG across a downstream switch can have at most a total of eight member links.

The `no` form of this command removes a VSX LAG.

Command context

```
config
```

Parameters

<LAG-ID>

Specifies the LAG ID. Required. Run the `show capacities` command for the maximum number of VSX LAGs supported for your particular type of switch.

Authority

Administrators

Usage

A VSX LAG across a VSX device can have at most a total of eight interfaces.

NOTE: When creating a VSX LAG, select an equal number of member links in each segment for load balancing, such as four member links (one segment) and four member links (another segment). Do not create a VSX LAG with six member links in one switch and two member links on another segment.

Example

Configuring LAG 100 as a VSX LAG:

```
switch(config)# interface lag 100 multi-chassis
```

Removing LAG 100 as a VSX LAG:

```
switch(config)# no interface lag 100
```

inter-switch-link {<PORT-NUM> | lag <LAG-ID>}

Syntax

```
inter-switch-link {<PORT-NUM> | lag <LAG-ID>}
```

```
no inter-switch-link
```

Description

Configures a physical port or a LAG as an interswitch link port. Only one port or LAG can be configured to act as an ISL.

The `no` form of this command clears the configuration of the interswitch link port from a physical port or a LAG.

NOTE: This `inter-switch-link {<PORT-NUM> | lag <LAG-ID>}` command replaces the `mclag inter-switch-link {<PORT-NUM> | lag <LAG-ID>}` command that is supported only in 10.00 releases.

Command context

`config-vsx`

Parameters

<PORT-NUM>

Specifies a physical port on the switch. Some switches (such as the Aruba 8400 Switch Series), use the format member/slot/port (for example, 1/1/1). Other switches use slot/port (for example, 1/1). Sets the port to act as ISL

<LAG-ID>

Specifies the LAG ID. Run the `show capacities` command for the maximum number of VSX LAGs supported for your particular type of switch.

Authority

Administrators

Examples

Configuring port 1/1/1 as an interswitch link port:

```
switch(config-vsx)# inter-switch-link 1/1/1
```

Configuring LAG 100 as an interswitch link port:

```
switch(config-vsx)# inter-switch-link lag 100
```

Clears the interswitch link port:

```
switch(config-vsx)# no inter-switch-link
```

inter-switch-link dead-interval

Syntax

```
inter-switch-link dead-interval <DEAD-INTERVAL>

no inter-switch-link dead-interval
```

Description

Sets the dead interval for the interswitch link protocol. The dead interval is the amount of time to wait for hellos from a peer before declaring the peer to be dead. The default dead interval time is 20 seconds.

The `no` form of this command resets the interswitch link dead interval to the default of 20 seconds.

NOTE: This `inter-switch-link dead-interval` command replaces the `mclag inter-switch-link dead-interval <DEAD-INTERVAL>` command that is supported only in 10.00 releases.

Command context

`config-vsx`

Parameters

<DEAD-INTERVAL>

Specifies the dead interval in seconds. Required. Range: 2 to 20 seconds.

Authority

Administrators

Examples

Setting the dead interval for the interswitch link protocol to 10 seconds:

```
switch(config)# vsx
switch(config-vsx)# inter-switch-link dead-interval 10
```

Setting the dead interval for the interswitch link protocol to the default:

```
switch(config)# vsx
switch(config-vsx)# no vsx inter-switch-link dead-interval
```

inter-switch-link hello-interval

Syntax

```
inter-switch-link hello-interval <HELLO-INTERVAL>
no inter-switch-link hello-interval
```

Description

Configures the interswitch link hello-interval. The hello interval determines the frequency of a hello packet exchange to confirm the control plane of the peer is alive. The default hello-interval is 1 second.

The `no` form of this command sets the interswitch link hello-interval to the default of 1 second.

NOTE: This `inter-switch-link hello-interval` command replaces the `mclag inter-switch-link hello-interval <HELLO-INTERVAL>` command that is supported only in 10.00 releases.

Command context

`config-vsx`

Parameters

<HELLO-INTERVAL>

Specifies hello interval in seconds. Range: 1 to 5 seconds.

Authority

Administrators

Examples

Configuring the interswitch link hello-interval to 3 seconds:

```
switch(config)# vsx
switch(config-vsx)# inter-switch-link hello-interval 3
```

Resetting the interswitch link hello-interval to the default of 1 second:

```
switch(config)# vsx
switch(config-vsx)# no inter-switch-link hello-interval
```

inter-switch-link hold-time

Syntax

```
inter-switch-link hold-time <HOLD-INTERVAL>
no inter-switch-link hold-time
```

Description

Sets the holdtime for the interswitch link protocol. A port is treated as down only when it stays down for the configured holdtime interval. The default holdtime is 0 seconds.

The `no` form of this command sets the interswitch link protocol holdtime to the default of 0 seconds.

NOTE: This `inter-switch-link hold-time` command replaces the `mclag inter-switch-link hold-time <HOLD-INTERVAL>` command that is supported only in 10.00 releases.

Command context

`config-vsx`

Parameters

<HOLD-INTERVAL>

Specifies the hold interval in seconds. Required. Range: 0 to 3 seconds.

Authority

Administrators

Examples

Setting the holdtime for interswitch link protocol to 2 seconds:

```
switch(config)# vsx
switch(config-vsx)# inter-switch-link hold-time 2
```

Setting the interswitch link protocol holdtime to the default of 0 seconds:

```
switch(config)# vsx
switch(config-vsx)# no inter-switch-link hold-time
```

keepalive dead-interval

Syntax

`keepalive dead-interval <DEAD-INTERVAL>`

`no keepalive dead-interval`

Description

Sets the dead-interval for keepalive protocol. The dead interval is the amount of time to wait for hellos from a peer before declaring the peer to be dead. The default dead-interval is 3 seconds.

The `no` form of this command sets the interswitch link dead-interval to the default of 3 seconds.

NOTE: This `keepalive dead-interval` command replaces the `mclag keepalive dead-interval <DEAD-INTERVAL>` command that is supported only in 10.00 releases.

Command context

`config-vsx`

Parameters

`dead-interval` ***<DEAD-INTERVAL>***

Specifies the dead-interval in seconds. Range: 2 to 20 seconds

Authority

Administrators

Examples

Setting the dead-interval for keepalive protocol to 10 seconds:

```
switch(config)# vsx
switch(config-vsx)# keepalive dead-interval 10
```

Setting the dead-interval for keepalive protocol to the default:

```
switch(config)# vsx
switch(config-vsx)# no keepalive dead-interval
```

keepalive hello-interval

Syntax

```
keepalive hello-interval <HELLO-INTERVAL>
```

```
no keepalive hello-interval
```

Description

Sets the hello-interval for keepalive protocol. The hello interval determines the frequency of a hello packet exchange to confirm the peer is alive. The default hello-interval is 1 second.

The `no` form of this command sets the hello-interval for keepalive protocol to the default of 1 second.

NOTE: This `keepalive hello-interval` command replaces the `mclag keepalive hello-interval <HELLO-INTERVAL>` command that is supported only in 10.00 releases.

Command context

```
config-vsx
```

Parameters

`hello-interval <HELLO-INTERVAL>`

Specifies the hello-interval in seconds. Range: 1 to 5 seconds

Authority

Administrators

Examples

Setting the hello-interval for keepalive protocol to 3 seconds:

```
switch(config)# vsx
switch(config-vsx)# keepalive hello-interval 3
```

Resetting the hello-interval for keepalive protocol to the default:

```
switch(config)# vsx
switch(config-vsx)# no keepalive hello-interval
```

keepalive peer

Syntax

`keepalive peer <PEER-IP-ADDR> source <SOURCE-IP-ADDR> [vrf <VRF-NAME>]`

`no keepalive`

Description

Sets the source and peer IP addresses for keepalive packets in a specified VRF. If a VRF is not specified, it sets to the default VRF.

The `no` form of this command removes the source and peer IP addresses and VRF for the keepalive protocol. VSX continues to work.

NOTE: This `keepalive peer` command replaces the `mclag keepalive peer <PEER-IP-ADDR> source <SOURCE-IP-ADDR> [vrf <VRF-NAME>]` command that is supported only in 10.00 releases.

Command context

`config-vsx`

Parameters

`peer <PEER-IP-ADDR>`

Specifies the peer IPv4 address. Syntax: A.B.C.D

`source <IP-ADDR>`

Specifies the source IPv4 address. The source IP address is the IP address assigned to the keepalive interface on the switch. For example, if you are entering this command on the primary switch, the source IP address would be the IP address assigned to the keepalive interface on the primary switch. Syntax: A.B.C.D

`vrf <VRF-NAME>`

Specifies the VRF name. If you are entering this command on the primary switch, the peer IP address is the IP address assigned to the keepalive interface for the secondary switch. If you are entering this command on the secondary switch, the peer IP address is the IP address assigned to the keepalive interface for the primary switch. Syntax: String

Authority

Administrators

Usage

To configure the keepalive feature, enter this command once on the primary switch and once on the secondary switch. The keepalive feature is recommended for redundancy. If the ISL link goes down, the keepalive connection keeps the traffic moving so that the peer and secondary switches can continue to communicate. The keepalive connection is established over a routed network, and it does not have to be a dedicated peer-to-peer link unlike ISL.

Examples

Setting the source and peer IP addresses for keepalive in the default VRF:

```
switch(config)# vsx
switch(config-vsx)# keepalive peer 192.168.1.1 source 192.168.1.5
```

Setting the source and peer IP addresses for keepalive in the vrf1:

```
switch(config)# vsx
switch(config-vsx)# keepalive peer 10.0.0.1 source 10.0.0.2 vrf vrf1
```

Removing the source and peer IP addresses and VRF for the keepalive protocol:

```
switch(config)# vsx
switch(config-vsx)# no keepalive
```

keepalive udp-port

Syntax

```
keepalive udp-port <PORT-NUM>
```

```
no keepalive udp-port
```

Description

Sets the UDP port for the keepalive protocol.

The `no` form of this command sets the UDP port for keepalive protocol to the default of 7678.

NOTE: This `keepalive udp-port` command replaces the `mclag keepalive udp-port <PORT-NUM>` command that is supported only in 10.00 releases.

Command context

```
config-vsx
```

Parameters

`udp-port <PORT-NUM>`

Specifies UDP port number. Range: 1024-65535

Authority

Administrators

Examples

Setting the UDP port for keepalive protocol to 2000:

```
switch(config)# vsx
switch(config-vsx)# vsx keepalive udp-port 2000
```

Setting the UDP port for keepalive protocol to the default of 7678:

```
switch(config)# vsx
switch(config-vsx)# no keepalive udp-port
```

linkup-delay-timer

Syntax

```
linkup-delay-timer <DELAY-TIMER>
```

```
no linkup-delay-timer
```

Description

Configures the VSX link-up delay timer. VSX delay timer feature lets you configure the delay timer, which delays bringing downstream VSX links up, following a VSX device reboot or an ISL flap. The linkup delay timer might need to be set to a higher value for larger networks, depending on the ARP and routing table size.

The `no` form of this command restores the VSX link-up delay timer to a default of 180 seconds.

Command context

`config-vsx`

Parameters

<DELAY-TIMER>

Specifies the VSX LAG bring-up delay in seconds. Range: 0 to 600 seconds

Authority

Administrators

Usage

Table 3: *Recommended delay timer settings for 8320 series switches*

MAC	ARPV4	Routes	Recommended delay timer setting
16K	16K	10K	120
32K	32K	10K	120
47K	47K	10K	150
47K	47K	10K	250
47K	47K	10K	250
47K	69K	10K OSPF	420

Table 4: *Recommended delay timer settings for 8400 series switches*

MAC	ARPV4	Routes	Recommended delay timer setting
40K	40K	512	300
32K	32K	512	180
48K	48K	512	480
48K	48K	20K IPv4 + 20K IPV6	600
48K	48K	10K IPv4 + 10K IPv6	480
32K	-	10K IPv4	180

Examples

Setting the VSX link-up delay timer to 35 seconds:

```
switch(config)# vsx
switch(config-vsx)# linkup-delay-timer 35
```

Setting the VSX link-up delay timer to the default:

```
switch(config)# vsx
switch(config-vsx)# no linkup-delay-timer
```

linkup-delay-timer exclude lag-list

Syntax

```
linkup-delay-timer exclude lag-list <LAG-LIST>
```

```
no linkup-delay-timer exclude lag-list <LAG-LIST>
```

Description

Configures the VSX link-up delay timer exclude list. It excludes the bringing up of specified downstream VSX LAGs, following a device reboot or an ISL flap.

The `no` form of this command unconfigures the VSX link-up delay timer exclude list.

Command context

```
config-vsx
```

Parameters

<LAG-LIST>

Specifies the LAGs to exclude. For example:

- `1`
- `1-10`
- `1,2,3`
- `1,2-10`

Authority

Administrators

Examples

Specifying LAGs to exclude LAG 100:

```
switch(config)# vsx
switch(config-vsx)# linkup-delay-timer exclude lag-list 100
```

Unconfiguring the VSX link-up delay timer exclude list for LAG 100:

```
switch(config)# vsx
switch(config-vsx)# no linkup-delay-timer exclude lag-list 100
```

role {primary | secondary}

Syntax

```
role {primary | secondary}

no role
```

Description

Configures the VSX device role.

The `no` form of this command removes the device role of the switch in VSX and causes the interswitch link to be out-of-sync.

Command context

```
config-vsx
```

Parameters

`{primary | secondary}`

Selects the VSX role to either primary or secondary for the device.

Authority

Administrators

Usage

VSX has no default role defined for the device. The device role assigns the device as the primary or secondary for VSX synchronization. For ISL to be in-sync, one device in VSX must be configured as the primary and the other device must be configured as the secondary.

Examples

Setting the VSX role to primary:

```
switch(config)# vsx
switch(config-vsx)# role primary
```

Removing the device role:

```
switch(config)# vsx
switch(config-vsx)# no role
```

show interface <VLAN-NAME>

Syntax

```
show interface <VLAN-NAME> [vsx-peer]
```

Description

Displays a virtual IPv4/IPv6 and MAC configured for active-active routing.

Command context

Operator (>) or Manager (#)

Parameters

<VLAN-NAME>

Specifies the VLAN name. Syntax: string

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed.

Authority

Operators or Administrators. Users without administrator authority can execute this command from the operator context (>) only.

Example

```
switch# show int vlan100

Interface vlan100 is up
 Admin state is up
 Hardware: Ethernet, MAC Address: 48:0f:cf:af:c1:9e
 IPv4 address 192.168.1.1/24
 active gateway 192.168.1.253 01:01:01:01:01:02
 active gateway fe80::00 01:01:01:01:01:02
 RX
       L3:
            ucast: 0 packets, 0 bytes
            mcast: 8 packets, 812 bytes
 TX
       L3:
            ucast: 2 packets, 80 bytes
            mcast: 0 packets, 0 byte
```

show lacp aggregates

Syntax

```
show lacp aggregates [<LAG-NAME>] [vsx-peer]
```

Description

Displays a specified LAG or all configured LAGs along with VSX LAGs.

Command context

Operator (>) or Manager (#)

Parameters

<LAG-NAME>

Specifies the LAG name. Optional. Syntax: string

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed.

Authority

Operators or Administrators. Users without administrator authority can execute this command from the operator context (>) only.

Example

Displaying all configured LAGs along with VSX LAGs:

```
switch# show lacp aggregates

Aggregate-name    : lag7 (multi-chassis)
Interfaces        : 1/1/1
Remote-interfaces : 1/1/1
Heartbeat rate    : slow
Hash              : l3-src-dst
Aggregate mode    : active

Aggregate-name    : lag9 (multi-chassis)
Interfaces        : 1/1/3
Remote-interfaces : 1/1/3
Heartbeat rate    : slow
Hash              : l3-src-dst
Aggregate mode    : active
```

Displaying a specified LAG:

```
switch# show lacp aggregates lag100
Aggregate-name        : lag100 (multi-chassis)
Interfaces            : 1/1/44
Remote-interfaces     : 1/1/44
Heartbeat rate        : slow
Hash                  : l3-src-dst
Aggregate mode        : active
```

show lacp interfaces multi-chassis

Syntax

```
show lacp interfaces multi-chassis [<IFNAME>] [vsx-peer]
```

Description

Shows all configured VSX remote interface details. The interface that has the ALFNCD status has been synced with the partner and is ready for flow distribution.

Command context

Operator (>) or Manager (#)

Parameters

<IFNAME>

Specifies the VSX interface name. Optional.

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed.

Authority

Operators or Administrators. Users without administrator authority can execute this command from the operator context (>) only.

Examples

```
switch# show lacp interfaces multi-chassis

State abbreviations :
A - Active         P - Passive      F - Aggregable I - Individual
S - Short-timeout L - Long-timeout N - InSync     O - OutofSync
C - Collecting     D - Distributing
X - State m/c expired               E - Default neighbor state

Actor details of all interfaces:
------------------------------------------------------------------------------
Intf Aggregate Port     Port     State   System-id         System   Aggr
     name      id       Priority                           Priority Key
------------------------------------------------------------------------------
2    lag100(mc)  1012    1        ALFNCD  48:0f:cf:af:13:b9 65534    100


Partner details of all interfaces:
------------------------------------------------------------------------------
Intf Aggregate Partner Port     State   System-id         System   Aggr
     name      Port-id Priority                           Priority Key
------------------------------------------------------------------------------
2    lag200(mc)  16      1        ALFNCD  48:0f:cf:af:84:d9   65534    200


Remote Actor details of all interfaces:
------------------------------------------------------------------------------
Intf Aggregate Port     Port     State   System-id         System   Aggr
     name      id       Priority                           Priority Key
------------------------------------------------------------------------------
2    lag100(mc)  1012   1        ALFOE   48:0f:cf:af:13:b9   65534    100


Remote Partner details of all interfaces:
------------------------------------------------------------------------------
Intf Aggregate Partner Port     State   System-id         System   Aggr
     name      Port-id Priority                           Priority Key
------------------------------------------------------------------------------
2    lag200(mc)  12      1       ALFOE   48:0f:cf:af:84:d9   65534   200
```

show vsx active-forwarding

Syntax

```
show vsx active-forwarding [interface <INTERFACE-VLAN>] [vsx-peer]
```

Description

Shows all the VSX active-forwarding configured interface VLANs or the VSX active-forwarding peer information for a particular interface VLAN.

Command context

Operator (>) or Manager (#)

Parameters

interface *<INTERFACE-VLAN>*

Specifies the interface VLAN name. Syntax: string

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed.

Authority

Operators or Administrators. Users without administrator authority can execute this command from the operator context (>) only.

Example

Displaying a list of VSX active-forwarding enabled interfaces:

```
switch# show vsx active-forwarding
List of VSX active-forwarding enabled interfaces:
vlan30
vlan32
vlan33
```

Displaying the VSX active-forwarding peer information for `vlan30`:

```
switch# show vsx active-forwarding interface vlan30
Interface vlan30 has VSX active-forwarding enabled.
Interface vlan30 Peer Data:
Peer MAC: 94:f1:28:21:22:00
Peer IPv6 Addresses:
    fe80::96f1:28ff:fe21:2200
```

show vsx brief

Syntax

```
show vsx brief [vsx-peer]
```

Description

Displays the brief VSX status.

NOTE: This `show vsx brief` command replaces the `show mclag brief` command that is supported only in 10.00 releases.

Command context

Operator (>) or Manager (#)

Parameters

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed.

Authority

Operators or Administrators. Users without administrator authority can execute this command from the operator context (>) only.

Example

Displaying the brief VSX status for the switch you are logged into:

```
vsx-primary# show vsx brief
ISL State                              : In-Sync
Keepalive State                        : Keepalive-Established
Device Role                            : primary
Number of Multi-chassis LAG interfaces : 2
```

Displaying the brief VSX status for the peer (secondary) switch while entering the command on the primary switch:

```
vsx-primary# show vsx brief
ISL State                              : In-Sync
Keepalive State                        : Keepalive-Established
Device Role                            : secondary
Number of Multi-chassis LAG interfaces : 2
```

Displaying the brief VSX status for the peer (primary) switch while entering the command on the secondary switch:

```
vsx-secondary# show vsx brief
ISL State                              : In-Sync
Keepalive State                        : Keepalive-Established
Device Role                            : primary
Number of Multi-chassis LAG interfaces : 2
```

show vsx config-consistency

Syntax

```
show vsx config-consistency [vsx-peer]
```

Description

Displays the VSX global configuration consistency between two VSX switches.

NOTE: This `show vsx config-consistency` command replaces the `show mclag config-consistency` command that is supported only in 10.00 releases.

Command context

Operator (>) or Manager (#)

Parameters

`[vsx-peer]`

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed.

Authority

Operators or Administrators. Users without administrator authority can execute this command from the operator context (>) only.

Example

```
switch# show vsx config-consistency
Configurations                         Local                                  Peer
-------------------------------------------------------------------------------------
software version                       0.1.0                                  0.1.0
ISL hello interval                     1                                      1
ISL dead interval                      5                                      5
ISL hold interval                      0                                      0
```

```
Keepalive hello interval                1                                   1
Keepalive dead interval                 3                                   3
Keepalive UDP port                      7678                                7678

ISL VLAN list
1,100
Peer ISL VLAN list
1,10
Local VSX active-forwarding enabled interface-vlans
2 5-9
Peer VSX active-forwarding enabled interface-vlans
2 5-10
```

show vsx config-consistency lacp

Syntax

`show vsx config-consistency lacp [<LAG-NAME>] [vsx-peer]`

Description

Displays VSX LACP configuration consistency between two VSX switches.

NOTE: This `show vsx config-consistency lacp` command replaces the `show mclag config-consistency lacp [<LAG-NAME>]` command that is supported only in 10.00 releases.

Command context

Operator (>) or Manager (#)

Parameters

<LAG-NAME>

Specifies the LAG name. Optional. Syntax: string

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed.

Authority

Operators or Administrators. Users without administrator authority can execute this command from the operator context (>) only.

Example

```
switch# show vsx config-consistency lacp
Configurations                          Local                   Peer
-------------------------------------------------------------------------
Name                                    lag100                  lag100
Loop protect enabled                    false                   true
Hash scheme                             l2-src-dst-hash         l2-src-dst-hash
Qos cos override                        0                       0
Qos dscp override                       0                       0
Qos trust

VSX VLAN list
1
Peer VSX VLAN list
1,10
-------------------------------------------------------------------------
```

```
Name                                    lag111                  lag111
Loop protect enabled                    false                   false
Hash scheme                             l2-src-dst-hash         l2-src-dst-hash
Qos cos override                        0                       0
Qos dscp override                       0                       0
Qos trust
VSX VLAN list
1
Peer VSX VLAN list
1
------------------------------------------------------
```

show vsx configuration {inter-switch-link | keepalive}

Syntax

```
show vsx configuration {inter-switch-link | keepalive} [vsx-peer]
```

Description

Displays the ISL configuration or keepalive protocol configuration in VSX.

NOTE: This `show vsx configuration {inter-switch-link | keepalive}` command replaces the `show mclag configuration` command that is supported only in 10.00 releases.

Command context

Operator (>) or Manager (#)

Parameters

`{inter-switch-link | keepalive}`

Selects `inter-switch-link` or `keepalive`.

`inter-switch-link`

Displays the ISL configuration in VSX.

`keepalive`

Displays the keepalive protocol configuration in VSX.

`[vsx-peer]`

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed.

Authority

Operators or Administrators. Users without administrator authority can execute this command from the operator context (>) only.

Example

Displaying the ISL configuration in VSX:

```
switch# show vsx configuration inter-switch-link
Inter Switch Link   : 1/1/43
Hello Interval      : 1 Seconds
Dead Interval       : 20 Seconds
Hold Time           : 0 Seconds
```

```
Device Role          : primary
Multichassis LAGs    : lag 100
```

Displaying the keepalive protocol configuration in VSX:

```
switch# show vsx configuration keepalive
Keepalive Interface     : 1/1/1
Keepalive VRF           : test1
Source IP Address       : 192.168.1.1
Peer IP Address         : 192.168.1.2
UDP Port                : 7678
Hello Interval          : 1 Seconds
Dead Interval           : 3 Seconds
```

show vsx ip data-path

Syntax

```
show vsx ip data-path [<IP-ADDR> | <IP-ADDR>/<MASK>] [vrf <VRF-NAME>] [vsx-peer]
```

Description

Displays the datapath of the IPv4 route present on local and VSX peer devices.

Command context

Operator (>) or Manager (#)

Parameters

<IP-ADDR> | <IP-ADDR>/<MASK>]

Selects one of the following: *<IP-ADDR>* or *<IP-ADDR>/<MASK>*

<IP-ADDR>

Specifies the datapath for an IPv4 address based on the parameters provided.

<IP-ADDR>/<MASK>

Specifies the datapath for an IPv4 address and its specified subnet. Optional. Syntax: A.B.C.D/M

vrf *<VRF-NAME>*

Shows the IPv4 datapath for a specified VRF.

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed.

Authority

Administrators

Example

Displaying the datapath on a VSX switch for 192.0.2.0:

```
switch# show vsx ip data-path 192.0.2.0

IPv4 Data Path Information For 192.0.2.0

Local Device
------------
Route : 192.0.2.0/32
```

```
    Egress L3 Interface : 1/1/2
    Next Hop MAC Address    : 08:00:09:ea:d7:d1
    Egress Port    : 1/1/2

    Egress L3 Interface : 1/1/3
    Nexthop Hop MAC Address    : 08:00:09:8e:59:1d
    Egress Port    : 1/1/3

Peer Device
------------
Route : 192.0.2.0/32
    Egress L3 Interface : loopback1
```

Displaying the datapath on a VSX switch for 198.51.100.0/32:

```
switch# show vsx ip data-path 198.51.100.0/32

IPv4 Data Path Information For 198.51.100.0/32

Local Device
------------
Route : 198.51.100.0/32
    Egress L3 Interface : 1/1/4
```

Displaying the datapaths on a VSX switch for 198.51.100.1:

```
switch# show vsx ip data-path 198.51.100.1

IPv4 Data Path Information For 198.51.100.1

Local Device
------------
Route : 198.51.100.1/32
    Egress L3 Interface : 1/1/4

Peer Device
------------
Route : 198.51.100.0/24
    Egress L3 Interface : 1/1/2
    Next Hop MAC Address    : 08:00:09:db:21:e8
    Egress Port    : 1/1/2
```

show vsx ipv6 data-path

Syntax

```
show vsx ipv6 data-path [<IPv6-ADDR> | <IPv6-ADDR>/<MASK>] [vrf <VRF-NAME>] [vsx-peer]
```

Description

Displays the datapath of the IPv6 route on local and peer VSX devices.

Command context

Operator (>) or Manager (#)

Parameters

<IPV6-ADDR> | <IPV6-ADDR>/<MASK>]

Selects one of the following: *<IPV6-ADDR>* or *<IPV6-ADDR>/<MASK>*

<IPV6-ADDR>

Specifies the datapath for an IPv6 address based on the parameters provided.

<IPV6-ADDR>/<MASK>

Specifies the datapath for an IPv6 address and its specified subnet. Optional. Syntax: A.B.C.D/M

vrf *<VRF-NAME>*

Shows the IPv6 datapath for a specified VRF.

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed.

Authority

Administrators

Example

Displaying an IPv6 datapath on a VSX switch:

```
switch# show vsx ipv6 data-path 1000::

IPv6 Data Path Information For 1000::

Local Device
------------
Route : 1000::/64
    Egress L3 Interface : 1/1/2

Peer Device
------------
Route : 1000::/64
    Egress L3 Interface : 1/1/2
```

Displaying an IPv6 datapath on a VSX switch:

```
switch# show vsx ipv6 data-path 2000::
IPv6 Data Path Information For 2000::

Local Device
------------
Route : 2000::/64
    Egress L3 Interface : 1/1/2
    Next Hop MAC Address   : 08:00:09:0e:0c:1b
    Egress Port   : 1/1/2
```

Displaying IPv6 datapath for 3000::/64 on a VSX switch:

```
switch# show vsx ipv6 data-path 3000::/64
IPv6 Data Path Information For 3000::/64

Local Device
------------
Route : 3000::/64
    Egress L3 Interface : 1/1/2
    Next Hop MAC Address   : 08:00:09:0e:0c:1b
    Egress Port   : 1/1/2
```

show vsx ip route

Syntax

```
show vsx ip route [<IP-ADDR> | <IP-ADDR>/<MASK> | unique]
     [vrf <VRF-NAME> | all-vrfs] [vsx-peer]
```

Description

Displays a specified LAG or all configured LAGs along with VSX LAGs.

Command context

Operator (>) or Manager (#)

Parameters

<IP-ADDR> | <IP-ADDR>/<MASK> | unique]

Selects one of the following: *<IP-ADDR>*, *<IP-ADDR>/<MASK>* , or `unique`

<IP-ADDR>

Specifies the route information for an IPv4 address based on the parameters provided.

<IP-ADDR>/<MASK>

Specifies the route information for an IPv4 address and its specified subnet. Optional. Syntax: A.B.C.D/M

unique

Specifies routes that are present only on the primary switch or only on the secondary switch. The routes that are present on both the primary and secondary switch are excluded. Optional. Syntax string.

vrf *<VRF-NAME>* | all-vrfs

Selects the VRF name or all VRFs.

<VRF-NAME>

Shows the IPv4 route information for a specified VRF.

all-vrf

Shows the IPv4 route information for all VRFs.

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed.

Authority

Administrators

Example

Displaying IPv4 routes on a VSX switch:

```
switch# show vsx ip route

IPv4 Forwarding Routes

'[x/y]' denotes [distance/metric]

192.0.2.0/32, vrf default
    via  192.0.2.1,  [1/0],  static on vsx1
    via  192.0.2.2,  [1/0],  static on vsx2
```

Displaying IPv4 routes on a VSX switch:

```
switch# show vsx ip route

IPv4 Forwarding Routes

'[x/y]' denotes [distance/metric]

192.0.2.3/24, vrf default
    via  1/1/3,  [0/0],  connected on vsx1
    via  192.0.2.2,  [110/2],  ospf on vsx2
192.0.2.4/32, vrf default
    via  1/1/3,  [0/0],  local on vsx1
192.0.2.5/24, vrf default
    via  1/1/4,  [0/0],  connected on vsx1
    via  192.0.2.2,  [110/3],  ospf on vsx2
192.0.2.6/32, vrf default
    via  1/1/4,  [0/0],  local on vsx1
192.0.2.7/32, vrf default
    via  192.0.2.8,  [110/1],  ospf on vsx1
    via  192.0.2.1,  [110/1],  ospf on vsx1
    via  loopback1,  [0/0],  local on vsx2
```

Displaying IPv4 unique routes on a VSX switch:

```
switch# show vsx ip route unique

IPv4 Forwarding Routes

'[x/y]' denotes [distance/metric]

192.0.2.0/32, vrf default
    via  192.0.2.2,  [1/0],  static on vsx2
192.0.2.9/32, vrf default
    via  192.0.2.1,  [1/0],  static on vsx1
```

Displaying IPv4 routes on a VSX switch for 192.0.2.10:

```
switch# show vsx ip route 192.0.2.10

IPv4 Forwarding Routes

'[x/y]' denotes [distance/metric]

192.0.2.10/32, vrf default
    via  192.0.2.1,  [1/0],  static on vsx1
    via  192.0.2.2,  [1/0],  static on vsx2
```

show vsx ipv6 route

Syntax

```
show vsx ipv6 route [<IPv6-ADDR> | <IPv6-ADDR>/<MASK> | unique]
     [vrf <VRF-NAME> | all-vrfs] [vsx-peer]
```

Description

Displays a specified LAG or all configured LAGs along with VSX LAGs.

Command context

Operator (>) or Manager (#)

Parameters

<IPV6-ADDR> | <IPV6-ADDR>/<MASK> | unique]

Selects one of the following: *<IPV6-ADDR>*, *<IPV6-ADDR>/<MASK>* , or `unique`

<IPV6-ADDR>

Specifies the route information for an IPv4 address based on the parameters provided.

<IPV6-ADDR>/<MASK>

Specifies the route information for an IPv4 address and its specified subnet. Optional. Syntax: A.B.C.D/M

unique

Specifies routes that are present only on the primary switch or only on the secondary switch. The routes that are present on both the primary and secondary switch are excluded. Optional. Syntax string.

vrf *<VRF-NAME>* | all-vrfs

Selects the VRF name or all VRFs.

<VRF-NAME>

Shows the IPv4 route information for a specified VRF.

all-vrf

Shows the IPv4 route information for all VRFs.

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed.

Authority

Administrators

Example

Displaying IPv6 routes on a VSX switch:

```
switch# show vsx ipv6 route

IPv6 Forwarding Routes

'[x/y]' denotes [distance/metric]

1000::/64, vrf default
    via  1/1/2,  [0/0],  connected on vsx1
    via  1/1/2,  [0/0],  connected on vsx2
1000::1/128, vrf default
    via  1/1/2,  [0/0],  local on vsx1
```

Displaying IPv6 unique routes on a VSX switch:

```
switch# show vsx ipv6 route unique
IPv6 Forwarding Routes

'[x/y]' denotes [distance/metric]

1000::1/128, vrf default
    via  1/1/2,  [0/0],  local on vsx1
1000::2/128, vrf default
    via  1/1/2,  [0/0],  local on vsx2
3000::/64, vrf default
    via  1000::2,  [1/0],  static on vsx1
```

Displaying IPv6 routes on a VSX switch for 2000::/64:

```
switch# show vsx ipv6 route 2000::/64
IPv6 Forwarding Routes

'[x/y]' denotes [distance/metric]

2000::/64, vrf default
    via  1000::2,  [1/0],  static on vsx1
    via  1000::1,  [1/0],  static on vsx2
```

show vsx status

Syntax

```
show vsx status [inter-switch-link | keepalive | linkup-delay] [vsx-peer]
```

Description

Displays global VSX status or a specified status determined by the selected parameter.

NOTE: This `show vsx status` command replaces the `show mclag status` and `show mclag keepalive status` commands that are supported only in 10.00 releases.

Command context

Operator (>) or Manager (#)

Parameters

`[inter-switch-link | keepalive | linkup-delay]`

Selects one of the following: `inter-switch-link`, `keepalive`, or `linkup-delay`

`inter-switch-link`

Specifies the display of the ISL status in VSX.

`keepalive`

Specifies the display of the VSX keepalive protocol status.

`linkup-delay`

Specifies the display of the VSX link-up delay information, such as the:

- Configured link-up delay timer
- Initial sync status
- Delay timer status
- LAGs on which the delay timer is running
- Time remaining for the interfaces to be brought up

`[vsx-peer]`

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed.

Authority

Operators or Administrators. Users without administrator authority can execute this command from the operator context (>) only.

Example

Displaying the global VSX status:

```
switch# show vsx status
VSX Operational State
---------------------
  ISL channel             : In-Sync
  ISL mgmt channel        : operational
  Config Sync Status      : in-sync
  NAE                     : peer_unreachable
  HTTPS Server            : peer_unreachable

Attribute           Local                       Peer
------------        --------                    --------
ISL link            1/1/43                      1/1/43
ISL version         0                           0
System MAC          48:0f:cf:af:70:84           48:0f:cf:af:c2:84
Platform            8320                        8320
Software Version    TL.10.01.00004              TL.10.01.00004
Device Role         primary                     secondary
```

Displaying the ISL status in VSX:

```
switch# show vsx status inter-switch-link
State                   : In-Sync
Link Status             : up
Mgmt state              : operational

Inter-switch link Statistics
----------------------------
Hello Packets Tx        : 4572
Hello Packets Rx        : 4573
Data Packets Tx         : 80634
Data Packets Rx         : 80637
Mgmt Packets Tx         : 25946
Mgmt Packets Rx         : 25167
Mgmt Packet Drops       : 0
```

Displaying the VSX keepalive protocol status:

```
switch# show vsx status keepalive
Keepalive State           : Keepalive-Established
Last Established          : Thu Jun  8 09:03:01 2018
Last Failed               : Thu Jun  8 09:04:02 2018
Peer System Id            : 58:1f:cf:af:a0:84
Peer Device Role          : primary

Keepalive Counters
Keepalive Packets Tx      : 322
Keepalive Packets Rx      : 121
Keepalive Timeouts        : 0
Keepalive Packets Dropped : 14
```

Displaying the VSX link-up delay status while ARP/MAC VSX synchronization is in progress:

```
switch# show vsx status linkup-delay
Configured linkup delay-timer                                   :    180 seconds
Initial sync status                                             :    in-progress
```

```
Delay timer status                                                    :    waiting-to-start
Interfaces that will be brought up after delay timer expires          :    lag20,lag30-lag31
Time remaining before interfaces are brought up                       :
```

Displaying the VSX link-up delay status with ARP/MAC VSX synchronization completed with the delay timer running:

```
switch# show vsx status linkup-delay
Configured linkup delay-timer                                         :    180 seconds
Initial sync status                                                   :    completed
Delay timer status                                                    :    running
Interfaces that will be brought up after delay timer expires          :    lag20,lag30-lag31
Time remaining before interfaces are brought up                       :    1 minutes 22 seconds
```

Displaying the VSX link-up delay status with ARP/MAC VSX synchronization completed and the delay timer expired:

```
switch# show vsx status linkup-delay
Configured linkup delay-timer                                         :    180 seconds
Initial sync status                                                   :    completed
Delay timer status                                                    :    completed
Interfaces that will be brought up after delay timer expires          :
Time remaining before interfaces are brought up                       :
```

Displaying the global VSX status for the peer switch:

```
vsx-primary# show vsx status vsx-peer
VSX Operational State
---------------------
  ISL channel             : In-Sync
  ISL mgmt channel        : operational
  Config Sync Status      : in-sync
  NAE                     : peer_reachable
  HTTPS Server            : peer_reachable

Attribute           Local               Peer
------------        --------            --------
ISL link            lag1                lag1
ISL version         2                   2
System MAC          e0:07:1b:cb:72:e4   98:f2:b3:68:79:2e
Platform            8320                8320
Software Version    TL.10.01.0002B      TL.10.01.0002B
Device Role         secondary           primary
```

show vsx status config-sync

Syntax

```
show vsx status config-sync [vsx-peer]
```

Description

Displays VSX configuration synchronization status for peers. This command can be run from the primary or secondary peer to view the configuration synchronization state.

Command context

Operator (>) or Manager (#)

Parameters

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed.

Authority

Operators or Administrators. Users without administrator authority can execute this command from the operator context (>) only.

Example

```
switch# show vsx status config-sync
Admin State              : Enabled
Operational State        : Operational
Error State              : None
Recommended remediation  : N/A
Current Time             : Wed Jul 18 23:41:07 2018
Last Sync Time           : Wed Jul 18 23:38:26 2018
```

vsx

Syntax

```
vsx
```

```
no vsx
```

Description

Creates the VSX context on the switch.

The `no` form of this command deletes the VSX context on the switch and removes all VSX configuration settings.

Command context

```
config
```

Authority

Operators or Administrators. Users without administrator authority can execute this command from the operator context (>) only.

Example

Creating the VSX context on the switch:

```
switch# vsx
switch(config-vsx)#
```

Removing the VSX context and all VSX configuration settings from the switch:

```
switch(config-vsx)# no vsx
VSX configuration will be deleted.
Do you want to continue (y/n)? y
switch(config)#
```

vsx active-forwarding

Syntax

```
vsx active-forwarding
```

```
no vsx active-forwarding
```

Description

Configures VSX active-forwarding on an interface VLAN.

The `no` form of this command unconfigures VSX active-forwarding on a VLAN interface.

Command context

```
config-if-vlan
```

Authority

Operators or Administrators. Users without administrator authority can execute this command from the operator context (>) only.

Example

Attempting to enable VSX active-forwarding on an interface that has an active-gateway enabled:

```
switch# interface vlan 3
switch(config-if-vlan)# vsx active-forwarding
Active-forwarding cannot be enabled on an interface that has active-gateway enabled
```

Successfully enabling VSX active-forwarding:

```
switch# interface vlan 3
switch(config-if-vlan)# vsx active-forwarding
switch(config-vsx)#
```

Unconfiguring VSX active-forwarding:

```
switch# interface vlan 3
switch(config-if-vlan)# no vsx active-forwarding
switch(config-vsx)#
```

vsx-sync

Syntax

```
vsx-sync
```

Description

Enables VSX sync for the entire VLAN context, access list context, policy context, or class context to the secondary peer.

Command context

```
config
```

- The `access-list ip <ACL-NAME>` command takes you into the named ACL context where you enter the `vsx-sync` command for an access control list.
- The `class IP <CLASS-NAME>` command takes you into the named class context where you enter the `vsx-sync` command for a class.
- The `policy <POLICY-NAME>` command takes you into the named policy context where you enter the `vsx-sync` command for a policy.
- The `vlan <ID>` command takes you into the `config-vlan` context where you enter the `vsx-sync` command for a VLAN.

Authority

Administrators

Example

Enabling VSX sync for the VLAN context to the secondary peer:

```
switch(config)# vlan 1
switch(config-vlan-1)# vsx-sync
```

Enabling VSX sync for the entire access list context to the secondary peer:

```
switch(config)# access-list ip ITBoston
switch(config-acl-ip)# vsx-sync
```

Enabling VSX sync for the entire policy context to the secondary peer:

```
switch(config)# policy ITPaloAlto
switch(config-policy)# vsx-sync
```

Enabling VSX sync for the entire class context to the secondary peer:

```
switch(config)# class ip ITHouston
switch(config-class-ip)# vsx-sync
```

vsx-sync active-gateways

Syntax

```
vsx-sync active-gateways
```

Description

Enables VSX sync for one or more gateways associated with the interface.

Command context

`config`

The `interface vlan <ID>` command takes you into the `config-if-vlan` context where you enter the `vsx-sync active-gateways` command for a VLAN.

Authority

Administrators

Example

Enabling VSX sync for active gateways 10.10.10.10 23:24:25:26:27:28 and fd12:3456:789a:1::1 23:24:25:26:27:28 associated with VLAN 1:

```
switch(config)# interface vlan 1
switch(config-if-vlan)# vsx-sync active-gateways
switch(config-if-vlan)# active-gateway ip 10.10.10.10 23:24:25:26:27:28
switch(config-if-vlan)# active-gateway ip6 fd12:3456:789a:1::1 23:24:25:26:27:28
```

vsx-sync {[vlans access-lists] | [access-lists | vlans]}

Syntax

```
vsx-sync {[vlans access-lists] | [access-lists | vlans]}
```

Description

Enables VSX synchronization for all VLANs or access lists or both associated with an interface.

Command context

```
(config)
```

- The `interface <PORT-NUM>` command takes you into the `config-if` context where you enter the `vsx-sync {[vlans access-lists] | [access-lists | vlans]}` command.
- The `interface lag <ID>` command takes you into the `config-lag-if` context where you enter the `vsx-sync {[vlans access-lists] | [access-lists | vlans]}` command.

Parameters

`{[vlans access-lists] | [access-lists | vlans]}`

Specifies that all VLANs and access lists associated with an interface are to be synced for VSX.

`[access-lists | vlans]`

Selects either all access lists or VLANs associated with an interface to be synced for VSX.

Authority

Administrators

Example

Enabling VSX synchronization for VLANs associated with interface 1/1/1:

```
switch(config)# interface 1/1/1
switch(config-if)# vsx-sync vlans
```

Enabling VSX synchronization for access lists associated with interface 1/1/1:

```
switch(config)# interface 1/1/1
switch(config-if)# vsx-sync access-lists
```

Enabling VSX synchronization for VLANs and access lists associated with interface 1/1/1:

```
switch(config)# interface 1/1/1
switch(config-if)# vsx-sync vlans access-lists
```

Enabling VSX synchronization for VLANs associated with LAG 1:

```
switch(config)# interface lag 1
switch(config-lag-if)# vsx-sync vlans
switch(config-lag-if)# vlan trunk native 1
```

Enabling VSX synchronization for access lists associated with LAG 2:

```
switch(config)# interface lag 2
switch(config-lag-if)#vsx-sync access-lists
switch(config-lag-if)# apply access-list ip ITBoston in
```

Enabling VSX synchronization for VLANs and access lists associated with LAG 3:

```
switch(config)# interface lag 3
switch(config-lag-if)#vsx-sync vlans access-lists
switch(config-lag-if)# vlan trunk native 1
switch(config-lag-if)# apply access-list ip ITPaloAlto in
```

Networking Websites

Hewlett Packard Enterprise Networking Information Library

www.hpe.com/networking/resourcefinder

Hewlett Packard Enterprise Networking Software

www.hpe.com/networking/software

Hewlett Packard Enterprise Networking website

www.hpe.com/info/networking

Hewlett Packard Enterprise My Networking website

www.hpe.com/networking/support

Hewlett Packard Enterprise My Networking Portal

www.hpe.com/networking/mynetworking

Hewlett Packard Enterprise Networking Warranty

www.hpe.com/networking/warranty

General websites

Hewlett Packard Enterprise Information Library

www.hpe.com/info/EIL

For additional websites, see **Support and other resources**.

Accessing Hewlett Packard Enterprise Support

- For live assistance, go to the Contact Hewlett Packard Enterprise Worldwide website:

  http://www.hpe.com/assistance

- To access documentation and support services, go to the Hewlett Packard Enterprise Support Center website:

  http://www.hpe.com/support/hpesc

Information to collect

- Technical support registration number (if applicable)
- Product name, model or version, and serial number
- Operating system name and version
- Firmware version
- Error messages
- Product-specific reports and logs
- Add-on products or components
- Third-party products or components

Accessing updates

- Some software products provide a mechanism for accessing software updates through the product interface. Review your product documentation to identify the recommended software update method.
- To download product updates:

  Hewlett Packard Enterprise Support Center

  www.hpe.com/support/hpesc

  Hewlett Packard Enterprise Support Center: Software downloads

  www.hpe.com/support/downloads

  Software Depot

  www.hpe.com/support/softwaredepot

- To subscribe to eNewsletters and alerts:

  www.hpe.com/support/e-updates

- To view and update your entitlements, and to link your contracts and warranties with your profile, go to the Hewlett Packard Enterprise Support Center **More Information on Access to Support Materials** page:

www.hpe.com/support/AccessToSupportMaterials

IMPORTANT: Access to some updates might require product entitlement when accessed through the Hewlett Packard Enterprise Support Center. You must have an HPE Passport set up with relevant entitlements.

Customer self repair

Hewlett Packard Enterprise customer self repair (CSR) programs allow you to repair your product. If a CSR part needs to be replaced, it will be shipped directly to you so that you can install it at your convenience. Some parts do not qualify for CSR. Your Hewlett Packard Enterprise authorized service provider will determine whether a repair can be accomplished by CSR.

For more information about CSR, contact your local service provider or go to the CSR website:

http://www.hpe.com/support/selfrepair

Remote support

Remote support is available with supported devices as part of your warranty or contractual support agreement. It provides intelligent event diagnosis, and automatic, secure submission of hardware event notifications to Hewlett Packard Enterprise, which will initiate a fast and accurate resolution based on your product's service level. Hewlett Packard Enterprise strongly recommends that you register your device for remote support.

If your product includes additional remote support details, use search to locate that information.

Remote support and Proactive Care information

HPE Get Connected

www.hpe.com/services/getconnected

HPE Proactive Care services

www.hpe.com/services/proactivecare

HPE Proactive Care service: Supported products list

www.hpe.com/services/proactivecaresupportedproducts

HPE Proactive Care advanced service: Supported products list

www.hpe.com/services/proactivecareadvancedsupportedproducts

Proactive Care customer information

Proactive Care central

www.hpe.com/services/proactivecarecentral

Proactive Care service activation

www.hpe.com/services/proactivecarecentralgetstarted

Warranty information

To view the warranty information for your product, see the links provided below:

HPE ProLiant and IA-32 Servers and Options

www.hpe.com/support/ProLiantServers-Warranties

HPE Enterprise and Cloudline Servers

www.hpe.com/support/EnterpriseServers-Warranties

HPE Storage Products

www.hpe.com/support/Storage-Warranties

HPE Networking Products

www.hpe.com/support/Networking-Warranties

Regulatory information

To view the regulatory information for your product, view the *Safety and Compliance Information for Server, Storage, Power, Networking, and Rack Products*, available at the Hewlett Packard Enterprise Support Center:

www.hpe.com/support/Safety-Compliance-EnterpriseProducts

Additional regulatory information

Hewlett Packard Enterprise is committed to providing our customers with information about the chemical substances in our products as needed to comply with legal requirements such as REACH (Regulation EC No 1907/2006 of the European Parliament and the Council). A chemical information report for this product can be found at:

www.hpe.com/info/reach

For Hewlett Packard Enterprise product environmental and safety information and compliance data, including RoHS and REACH, see:

www.hpe.com/info/ecodata

For Hewlett Packard Enterprise environmental information, including company programs, product recycling, and energy efficiency, see:

www.hpe.com/info/environment

Documentation feedback

Hewlett Packard Enterprise is committed to providing documentation that meets your needs. To help us improve the documentation, send any errors, suggestions, or comments to Documentation Feedback (**docsfeedback@hpe.com**). When submitting your feedback, include the document title, part number, edition, and publication date located on the front cover of the document. For online help content, include the product name, product version, help edition, and publication date located on the legal notices page.