High Availability SSO Deployment Guide for Cisco Catalyst 9800 Series Wireless Controllers, Cisco IOS XE Amsterdam 17.3

First Published: August 5, 2020

Cisco Systems, Inc. www.cisco.com

Table of Contents

Introduction

High availability has been a requirement on wireless controllers to minimize downtime in live networks. This document provides information on the theory of operation and configuration for the Catalyst 9800 Wireless Controller as it pertains to supporting stateful switchover of access points and clients (AP and Client SSO). Catalyst 9800 Wireless Controller is the next generation wireless controller that can run on multiple platforms with different scalability goals from low to high scale. AP and Client SSO is supported on the physical appliances and the virtual cloud platforms of the Catalyst 9800 Wireless Controller, namely C9800-L, C9800-40, C9800-80 and C9800-CL. The underlying SSO functionality is the same on all platforms with some differences in the setup process.

Overview

The High availability SSO capability on wireless controller allows the access point to establish a CAPWAP tunnel with the Active wireless controller and the Active wireless controller to share a mirror copy of the AP and client database with the Standby wireless controller. The APs do not go into the Discovery state and clients do not disconnect when the Active wireless controller fails and the Standby wireless controller takes over the network as the Active wireless controller. There is only one CAPWAP tunnel maintained at a time between the APs and the wireless controller that is in an Active state.

Release 16.10 supports full access point and Client Stateful Switch Over. Client SSO is supported for clients which have already completed the authentication and DHCP phase and have started passing traffic. With Client SSO, a client's information is synced to the Standby wireless controller when the client associates to the wireless controller or the client's parameters change. Fully authenticated clients, i.e. the ones in Run state, are synced to the Standby and thus, client re-association is avoided on switchover making the failover seamless for the APs as well as for the clients, resulting in zero client service downtime and zero SSID outage. The overall goal for the addition of AP and client SSO support to the Catalyst 9800 Wireless controller is to reduce major downtime in wireless networks due to failure conditions that may occur due to box failover, network failover or power outage on the primary site.

Feature Description and Functional Behavior

All the control plane activities are centralized and synchronized between the active and standby units. The Active Controller centrally manages all the control and management communication. The network control data traffic is transparently switched from the standby unit to the active unit for centralized processing.

Bulk and Incremental configuration is synced between the two controllers at run-time and both controllers share the same IP address on the management interface. The CAPWAP state of the Access Points that are in Run State is also synched from the active wireless controller to the Hot-Standby wireless controller allowing the Access Points to be state-fully switched over when the Active wireless controller fails. The APs do not go to the Discovery state when Active wireless controller fails, and Standby wireless controller takes over as the Active wireless controller to serve the network.

The two units form a peer connection through a dedicated RP port (this can be a physical copper or fiber port) or a virtual interface for the VM. The Active/Standby election happens at boot time and it's either based on the highest priority (priority range is <1-2>) or the lowest MAC if the priority is the same. By default the C9800 has a priority of 1. Once the HA pair is formed, all the configuration and AP and client databases are synched between Active and standby. Any configuration is done on the Active is automatically synch to the Standby. The standby is continuously monitoring the Active via keepalives over the RP link. If the Active becomes unavailable, the standby assumes the role of Active. It does that by sending a Gratuitous ARP message advertising to the network that it now owns that wireless management IP address. All the configurations and databases are already in synch, so the standby can take over without service disruption

There is no pre-empt functionality with SSO meaning that when the previous Active wireless controller resumes operation, it will not take back the role as an Active wireless controller but will negotiate its state with the current Active wireless controller and transition to Hot-Standby state.

Platforms Supported

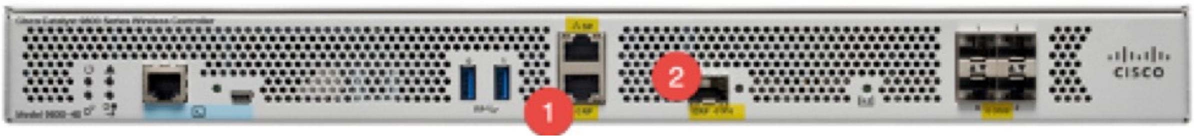
- Cisco Catalyst C9800-40 Wireless Controller
- Cisco Catalyst C9800-80 Wireless Controller
- Cisco Catalyst C9800-CL Wireless Controller
- Cisco Catalyst C9800-L Wireless Controller

SSO Pre-requisites

- HA Pair can only be form between two wireless controllers of the same form factor
- Both controllers must be running the same software version in order to form the HA Pair
- Maximum RP link latency = 80 ms RTT, minimum bandwidth = 60 Mbps and minimum MTU = 1500

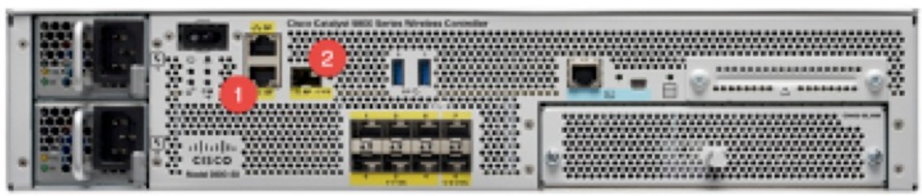
SSO on Cisco Catalyst C9800-40-K9 and C9800-80-K9 Wireless Controllers

The Cisco C9800-40-K9 wireless controller is an extensible and high performing wireless controller, which can scale up to 2000 access points and 32000 clients. The controller has four 10G data ports and a throughput of 40G.



1	RP– RJ-45 1G redundancy Ethernet port.	2	Gigabit SFP RP port

The Cisco C9800-80-K9 Wireless Controller is a 100G wireless controller that occupies two rack unit space and supports a pluggable Module slot, and eight built-in 10GE/1GE interfaces.



1	RP– RJ-45 1G redundancy Ethernet port.	2	Gigabit SFP RP port

Both C9800-40-K9 and C9800-80-K9 Wireless controllers have two RP Ports as shown in the figures above:

- RJ-45 Ethernet Redundancy port
- SFP Gigabit Redundancy Port

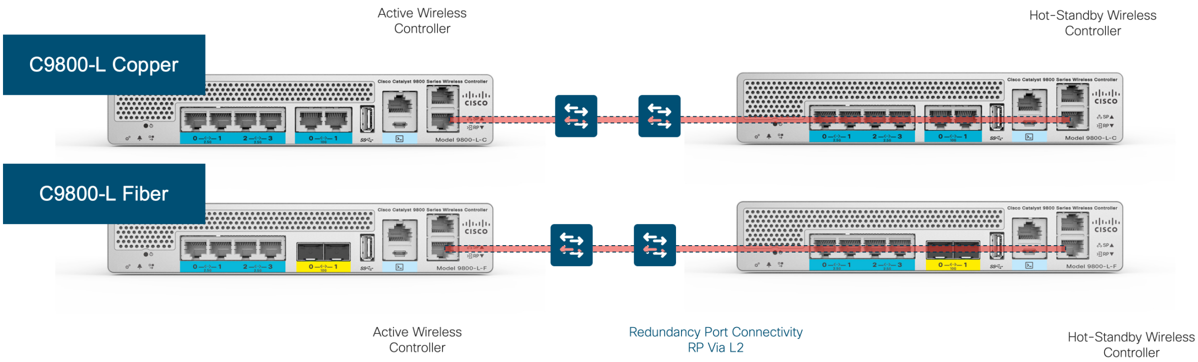
If both the Redundancy Ports are connected:

- SFP Gigabit Ethernet port takes precedence if they are connected at same time.
- HA between RJ-45 and SFP Gigabit RP ports is not supported.
- Only Cisco supported SFPs (GLC-LH-SMD and GLC-SX-MMD) are supported for RP port
- 10G SFP-10G-SR is not supported on the RP port.
- When HA link is up via RJ-45, SFPs on HA port should not be inserted even if there is no link between them. As it is a physical level detection, this would cause the HA to go down as precedence is given to SFP

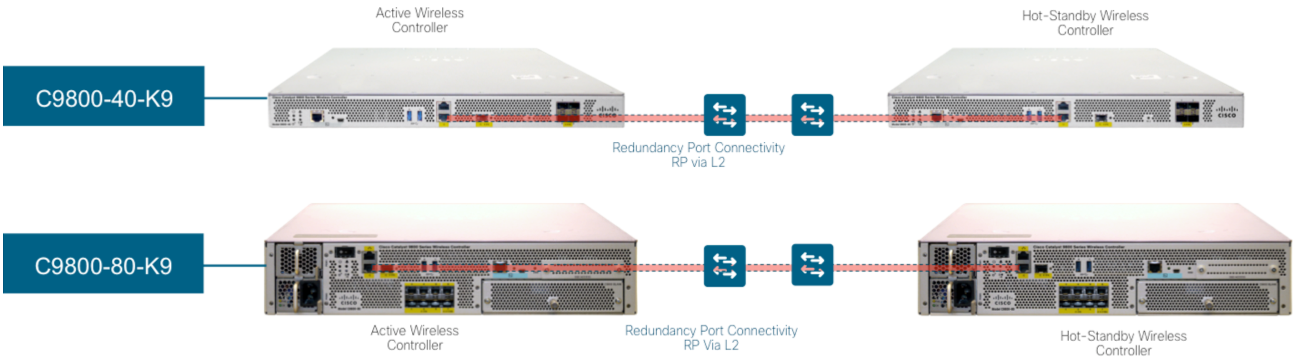
Physical Connectivity for C9800-L, C9800-40 and C9800-80 Wireless Controller HA SSO

The HA Pair always has one active controller and one standby controller. If the active controller becomes unavailable, the standby assumes the role of the active. The Active wireless controller creates and updates all the wireless information and constantly synchronizes that information with the standby controller. If the active wireless controller fails, the standby wireless controller assumes the role of the active wireless controller and continues to the keep the HA Pair operational. Access Points and clients continue to remain connected during an active-to-standby switchover.

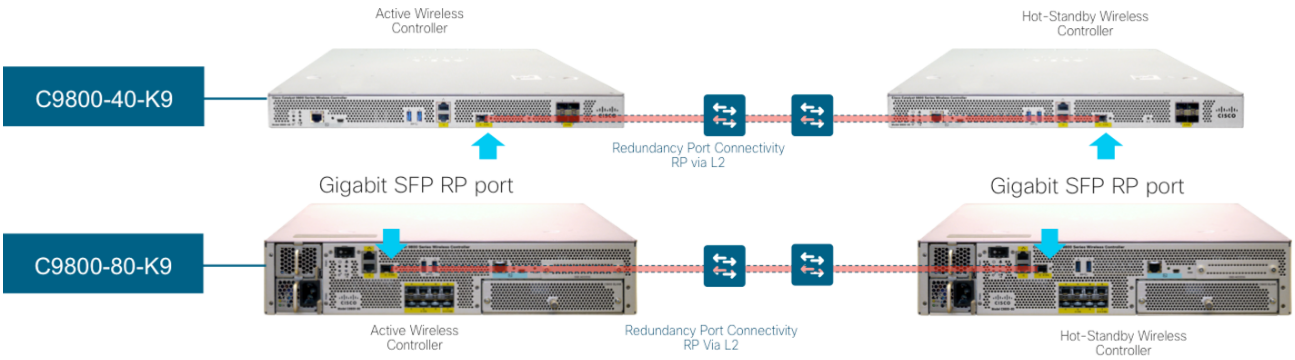
Connecting C9800-L Wireless Controllers using RJ-45 RP Port for SSO



Connecting C9800-40 and 9800-80 Wireless Controllers using RJ-45 RP Port for SSO



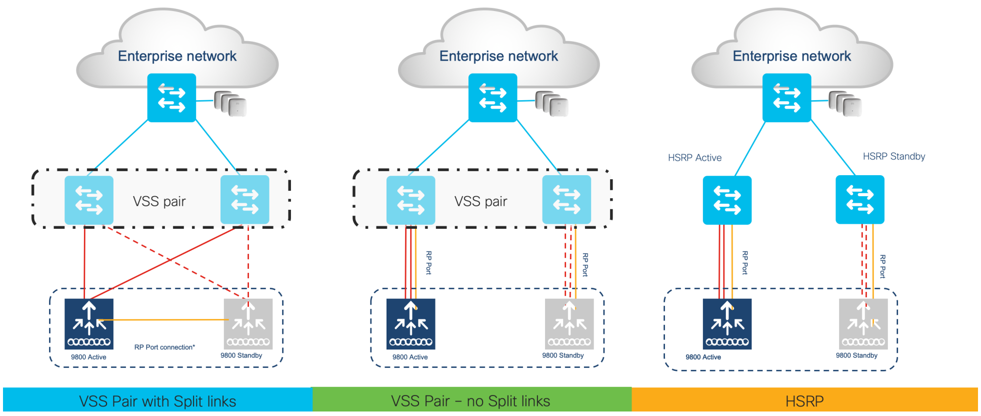
Connecting C9800-40 and 9800-80 Wireless Controllers using SFP Gigabit RP Port for SSO



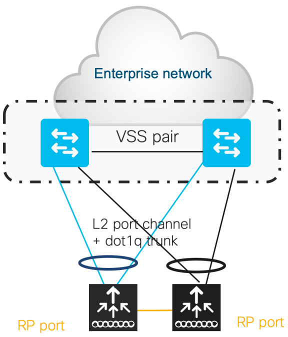
Connecting a C9800 wireless controller HA pair to upstream switches

Prior to 17.1 following topologies were supported in terms of upstream connectivity to the network:

1. SSO pair connected to upstream VSS pair with split links and RP connected back to back.
2. SSO pair connected to upstream VSS pair with RP connected via the upstream set of switches in order to detect gateway down scenario.
3. SSO pair connected to upstream HSRP active and standby and RP connected via upstream set of switches in order to detect gateway down scenario.



Option 1: Single VSS switch (or stack/VSL pair/modular switch) with RP back-to-back

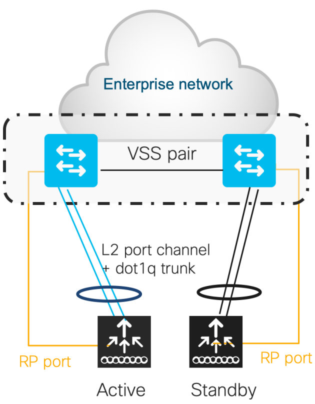


Single L2 port-channel on each box and enable dot1q to carry multiple VLANs. Spread the uplinks of the HA pair across the VSS pair and connect the RP back to back (no L2 network in between). Make sure that switch can scale in terms of ARP and MAC table entries.

This is a recommended topology.

Note: In HA SSO topology only LAG with mode ON is supported.

Option 2: Single VSS switch (or stack/VSL pair/modular switch) with RP via upstream

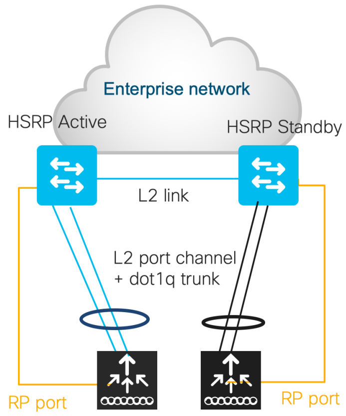


With this topology a single L2 port-channel is created on each box. Enable dot1q to carry multiple VLANs and connect the standby in the same manner. Make sure that switch can scale in terms of ARP and MAC table entries

IMPORTANT: In this topology the links are not spread across the VSS stack. Connect RP port to the same VSS/stack member as the uplinks and not back to back

Note: In HA SSO topology only LAG with mode ON is supported.

Option 3: Dual Distributed switches with HSRP



With this topology a single L2 port-channel is created on each box. Enable dot1q to carry multiple VLANs and connect the standby in the same manner. Make sure that switch can scale in terms of ARP and MAC table entries.

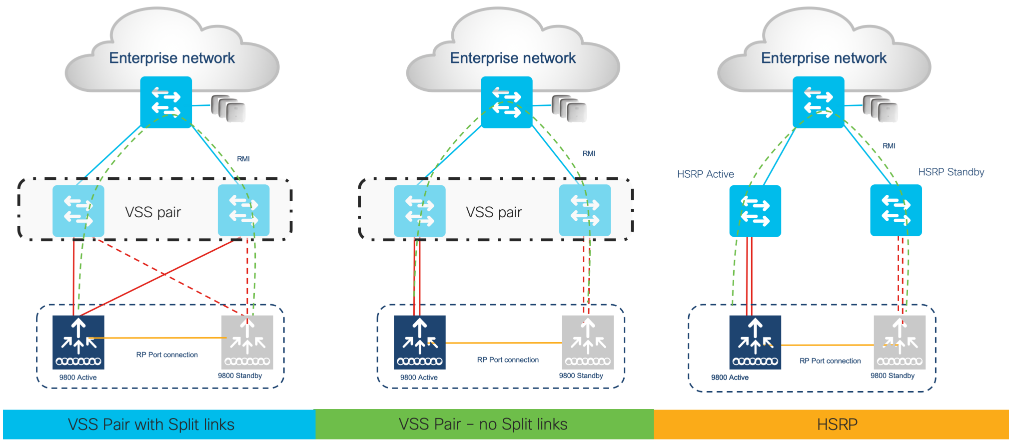
IMPORTANT: Connect RP port to the same distribution switch as the uplinks and not back to back

Note: In HA SSO topology only LAG with mode ON is supported prior to release 17.1. With 17.1, we additionally support LACP and PAGP. See the LACP, PAGP support in SSO Pair section for more details

Connecting a C9800 wireless controller HA pair to upstream switches with Release 17.1 and above

With the option of RMI and default gateway check feature available in release 17.1, the following topologies are now supported and recommended:

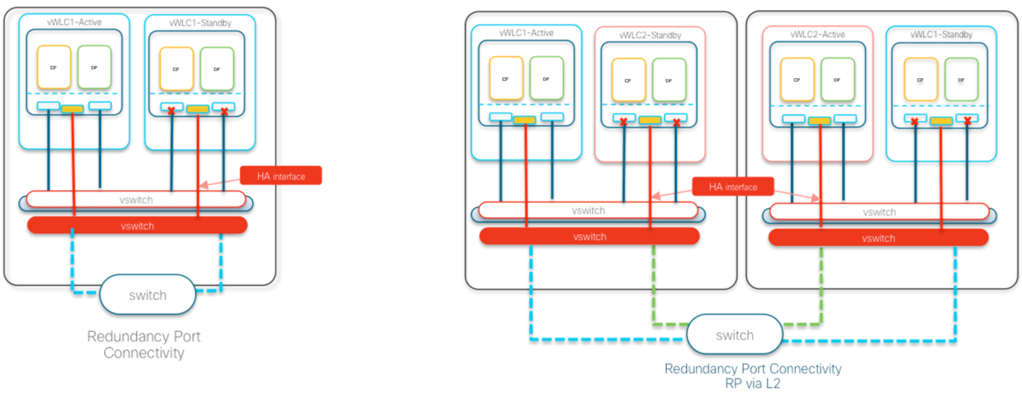
1. SSO pair connected to upstream VSS pair with split links and RP connected back to back.
2. SSO pair connected to upstream VSS pair and RP connected back to back.
3. SSO pair connected to upstream HSRP active and standby and RP connected back to back.



Note: It is recommended to configure portfast trunk in uplink switches for faster convergence using CLI "spanning-tree port type edge trunk" or "spanning-tree portfast trunk"

SSO on Cisco Catalyst C9800-CL running on ESXi, KVM, Hyper-V

The Virtual Catalyst 9800 Wireless controller can be deployed as an HA Pair in a single or dual server setup.



The figure on the left shows Redundant port connected on the same server.

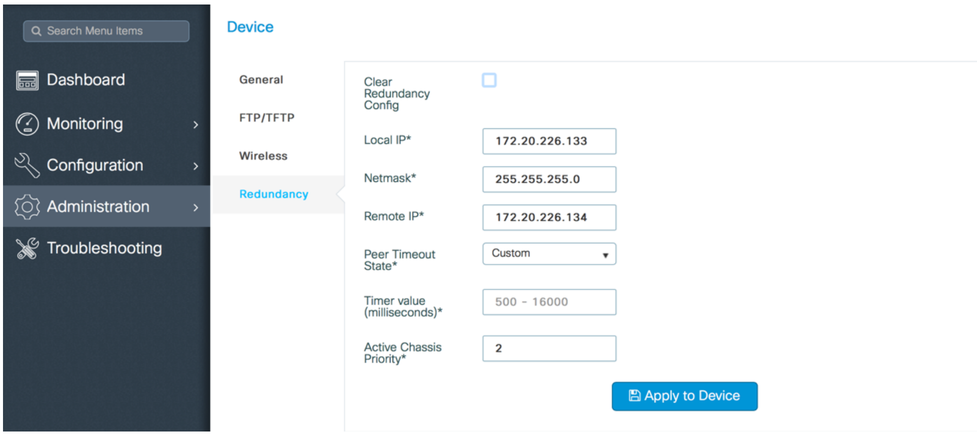
The figure on the right shows Redundant port L2 connected to a separate server.

The same interface number (for example Gig3) must be used to form the HA pair on 9800-CL. The scale of templates must also match. We support SSO across 9800-CL on HyperV, VMware ESXi and KVM.

Configuring High Availability SSO using GUI

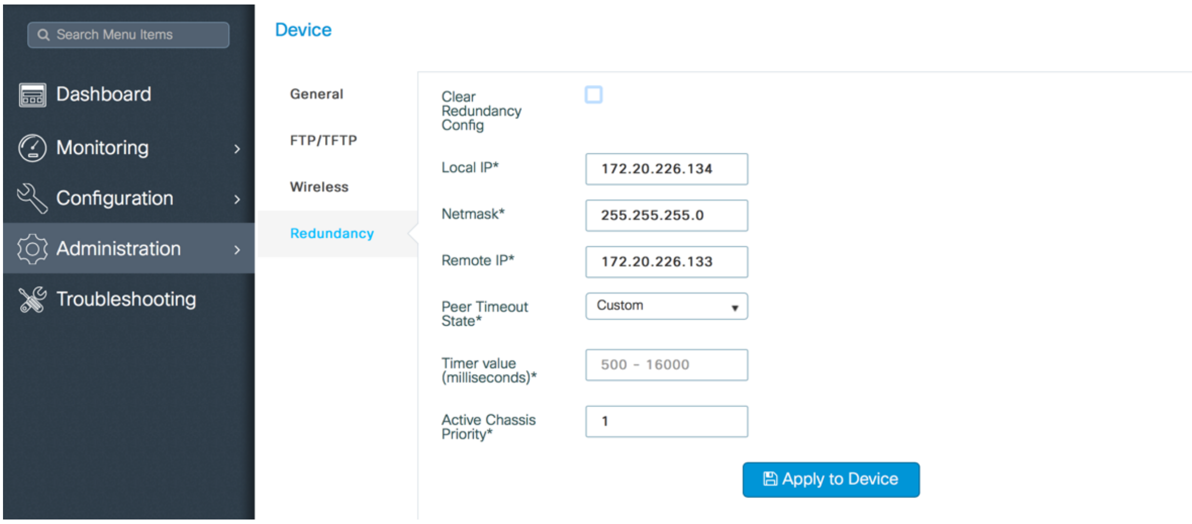
Device redundancy can be configured from **the Administration > Device > Redundancy** page

On the Active controller, the priority is set to a higher value than the standby controller. The wireless controller with the higher priority value is selected as the active during the active-standby election process. The Remote IP is the IP address of the standby controller's redundancy port IP.



Note: This page has changed starting release 17.1 to include an option to configure the HA pair using RMI. Please refer to the Redundancy Management Interface section to see the updated screens for configuration.

On the standby controller, the remote IP is set to the Active controller's redundancy port IP



Clear Redundancy config clears the SSO configuration and returns the controller to standalone mode.

Note: It is recommended to configure HA using the Redundancy Management Interface (RMI) starting Release 17.1. To see configuration using RMI please see the Redundancy Management Interface section.

Configuring High Availability SSO using CLI

▪ **On Virtual Catalyst 9800 Wireless controller**, enable High Availability SSO using the following command on each of the two virtual Catalyst 9800 Wireless controller instances

```
chassis redundancy ha-interface <RP interface> local-ip <local IP> <local IP subnet>
remote-ip  <remote IP>
```

e.g.

On Virtual Catalyst 9800 Wireless controller instance-1:

```
chassis redundancy ha-interface Gig 3 local-ip 172.23.174.85 /24 remote-
ip  172.23.174.86
```

On Virtual Catalyst 9800 Wireless controller instance-2:

```
chassis redundancy ha-interface Gig 3 local-ip 172.23.174.86 /24 remote-ip 172.23.174.85
```

■ **On C9800-40 and C9800-80 wireless controller,** enable High Availability SSO using the following command on each of the two wireless controller units

```
chassis redundancy ha-interface local-ip <local IP> <local IP subnet> remote-ip <remote IP>
```

Reload both wireless controllers by executing the command reload from the CLI

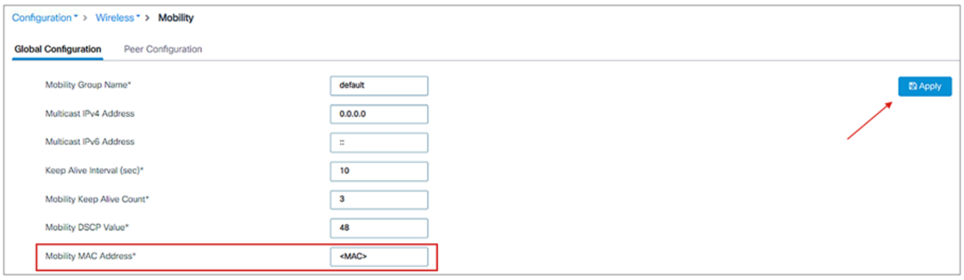
Note: It is recommended to configure HA using the Redundancy Management Interface (RMI) starting Release 17.1. To see configuration using RMI please see the Redundancy Management Interface section.

Mobility MAC

The wireless mobility MAC is the MAC address used for mobility communication. In an SSO scenario, ensure that you explicitly configure the wireless mobility MAC address; otherwise, the mobility tunnel will go down after SSO. The mobility MAC address for the SSO pair can be configured either:

- Before forming the SSO pair on each standalone controller. This is recommended before software release 16.12.3.
- On the active controller once the SSO pair is formed.

To configure the mobility MAC address, you can use the GUI:



Once you've entered the address, click Apply.

Note: The MAC address on the GUI is automatically derived from the wireless management interface, but you can use any other valid MAC address.

In the CLI, use the following command:

```
C9800#wireless mobility mac-address <MAC>
```

Active and Standby Election Process

An active C9800 wireless controller retains its role as an Active Controller unless one of the following events occur:

■ The wireless controller HA pair is reset.

- The active wireless controller is removed from the HA pair.
- The active wireless controller is reset or powered off.
- The active wireless controller fails.

The active wireless controller is elected or re-elected based on one of these factors and in the order listed below:

1. The wireless controller that is currently the active wireless controller.
2. The wireless controller with the highest priority value.

   Note: We recommend assigning the highest priority value to the wireless controller C9800 you prefer to be the active controller. This ensures that the controller is re-elected as active controller if a re-election occurs.

   Setting the Switch Priority Value

   ```
   chassis chassis -number priority new-priority-number
   ```

   Chassis-number Specifies the chassis number and the new priority for the chassis. The chassis number range is 1 to 2.

   The priority value range is <1-2>

   Example

   ```
   wireless controller#chassis 1 priority 2
   ```

   You can display the current priority value by using the **show chassis** user EXEC command. The new priority value takes effect immediately but does not affect the current Active Controller. The new priority value helps determine which controller is elected as the new Active Controller when the current active wireless controller or HA redundant pair reloads.

3. The wireless controller with the shortest start-up time.
4. The wireless controller with the lowest MAC Address.

The HA LED on the chassis can be used to identify the current Active Controller.

State Transition for HA SSO Pair formation

1. Active wireless controller in Non Redundant mode

```
TLV(0): T=9, L=29, V=KEY_TLV_PACKAGE_COMPATIBILITY
FRU Key detected
TLV(1): T=9, L=11, V=FRU_RP_TYPE
found package fru type FRU_RP_TYPE
TLV(2): T=9, L=24, V=KEY_TLV_PACKAGE_BOOTARCH
ARCH Key detected
TLV(3): T=9, L=14, V=ARCH_i686_TYPE
found package arch type ARCH_i686_TYPE
TLV(4): T=9, L=20, V=KEY_TLV_BOARD_COMPAT
TLV(5): T=9, L=15, V=BOARD_qwlc_TYPE
TLV(6): T=9, L=24, V=KEY_TLV_CRYPTO_KEYSTRING
TLV(7): T=9, L=4, V=none
TLV(8): T=9, L=11, V=CW_BEGIN=$$
TLV(9): T=9, L=16, V=CW_FAMILY=$qwlc$
TLV(10): T=9, L=78, V=CW_IMAGE=$qwlc-universalk9_wlc.BLD_POLARIS_DEV_LATEST_20180310_120257.SSA.bin$
TLV(11): T=9, L=19, V=CW_VERSION=$16.9.1$
TLV(12): T=9, L=52, V=CW_DESCRIPTION=$Cisco IOS Software, IOS-XE Software$
TLV(13): T=9, L=9, V=CW_END=$$
found DIGISIGN TLV type 12 length = 388

RSA Signed DEVELOPMENT Image Signature Verification Successful.
Validating subpackage signatures: addr=0x6e13e3f8, size=01c789ed

initramfs_size: 0x1c78dcd - 0x4b0a38 - 0x3e0 = 0x17c7fb5
Image validated
Booting image with bootparam="root=/dev/ram rw console=tty1 max_loop=64 pciehp.pciehp_force pcie_ports=native SR_BOOT=tftp://172.25.140.118/auto/
tftpboot/maahmed/qwlc-universalk9_wlc.BLD_POLARIS_DEV_LATEST_20180310_120257.SSA.bin rd_start=0xaf06e000 rd_size=0x17c7fb5 pkg_start=0x33f68000
pkg_size=0x3a1d4000 bdinfo_start=0xcd42b000 bdinfo_size=0x35c34"
May  3 15:13:22.585: %BOOT-0-DRV_LOADFAIL: R0/0: binos: Failed to load driver modprobe ( /usr/binos/conf/driver_common.sh: line 99: indigorw:
command not found )
May  3 15:13:43.295: %PMAN-3-PROC_EMPTY_EXEC_FILE: R0/0: pvp: Empty executable used for process bt_logger
May  3 15:13:45.742: %PMAN-3-PROC_EMPTY_EXEC_FILE: R0/0: pvp: Empty executable used for process bt_logger

Waiting for remote chassis to join
```

2. Standby Insertion for HA Pairing

```
Chassis number is 1
All chassis in the stack have been discovered. Accelerating discovery
May  3 15:13:46.276: %PMAN-3-PROC_EMPTY_EXEC_FILE: R0/0: pvp: Empty executable used for process bt_logger
May  3 15:13:46.877: %PMAN-3-PROC_EMPTY_EXEC_FILE: R0/0: pvp: Empty executable used for process bt_logger
May  3 15:13:48.852: %PMAN-3-PROC_EMPTY_EXEC_FILE: R0/0: pvp: Empty executable used for process bt_logger
May  3 15:13:53.654: %PMAN-3-PROC_EMPTY_EXEC_FILE: R0/0: pvp: Empty executable used for process bt_logger
May  3 15:13:56.934: %PMAN-3-PROC_EMPTY_EXEC_FILE: R0/0: pvp: Empty executable used for process bt_logger

              Restricted Rights Legend

Use, duplication, or disclosure by the Government is
subject to restrictions as set forth in subparagraph
(c) of the Commercial Computer Software - Restricted
Rights clause at FAR sec. 52.227-19 and subparagraph
(c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.

           Cisco Systems, Inc.
           170 West Tasman Drive
           San Jose, California 95134-1706
```

3. HA Sync in Progress

```
directory.
*May  3 15:13:52.681: %STACKMGR-6-STACK_LINK_CHANGE: Chassis 2 R0/0: stack_mgr: Stack port 2 on Chassis 1 is down
*May  3 15:13:52.681: %STACKMGR-6-STACK_LINK_CHANGE: Chassis 2 R0/0: stack_mgr: Stack port 1 on Chassis 1 is up
*May  3 15:13:52.681: %STACKMGR-6-STACK_LINK_CHANGE: Chassis 2 R0/0: stack_mgr: Stack port 2 on Chassis 1 is up
*May  3 15:13:52.682: %STACKMGR-6-CHASSIS_ADDED: Chassis 2 R0/0: stack_mgr: Chassis 2 has been added to the stack.
*May  3 15:13:52.682: %STACKMGR-6-CHASSIS_ADDED: Chassis 2 R0/0: stack_mgr: Chassis 2 has been added to the stack.
*May  3 15:13:52.682: %STACKMGR-6-ACTIVE_ELECTED: Chassis 2 R0/0: stack_mgr: Chassis 1 has been elected ACTIVE.
*May  3 15:13:52.682: %CMRP-3-PFU_MISSING: Chassis 2 R0/0: cmand: The platform does not detect a power supply in slot 1
*May  3 15:14:41.704: %SYS-4-FREEMEMWARNING: SIP0/0: Free Memory has dropped below warning threshold.
*May  3 15:14:46.405: %SYS-6-BOOTTIME: Time taken to reboot after reload = 1073 seconds
*May  3 15:14:46.761: %PNP-6-PNP_DISCOVERY_STOPPED: PnP Discovery stopped (Startup Config Present)
*May  3 15:14:46.789: %SPA_OIR-6-ONLINECARD: SPA (BUILT-IN-4X10G/1G) online in subslot 0/0
*May  3 15:14:46.883: %IOSXE_SPA-6-UPDOWN: Interface TenGigabitEthernet0/0/0, link down due to local fault
*May  3 15:14:46.937: %IOSXE_SPA-6-UPDOWN: Interface TenGigabitEthernet0/0/1, link down due to local fault
*May  3 15:14:46.977: %IOSXE_SPA-6-UPDOWN: Interface TenGigabitEthernet0/0/2, link down due to local fault
*May  3 15:14:47.040: %IOSXE_SPA-6-UPDOWN: Interface TenGigabitEthernet0/0/3, link down due to local fault
*May  3 15:14:48.780: %LINK-3-UPDOWN: Interface TenGigabitEthernet0/0/1, changed state to down
*May  3 15:14:48.783: %LINK-3-UPDOWN: Interface TenGigabitEthernet0/0/2, changed state to down
*May  3 15:14:48.784: %LINK-3-UPDOWN: Interface TenGigabitEthernet0/0/3, changed state to down
*May  3 15:14:49.217: %IOSXE_SPA-6-UPDOWN: Interface TenGigabitEthernet0/0/0, link down due to remote fault
*May  3 15:14:49.032: %LINK-3-UPDOWN: SIP0/0: Interface TenGigabitEthernet0/0/0, changed state to down
*May  3 15:14:49.652: %LINK-3-UPDOWN: Interface TenGigabitEthernet0/0/0, changed state to down
*May  3 15:14:50.043: %LINK-3-UPDOWN: Interface GigabitEthernet0, changed state to up
*May  3 15:14:51.043: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0, changed state to up
*May  3 15:14:54.229: %PKI-2-NON_AUTHORITATIVE_CLOCK: PKI functions can not be initialized until an authoritative time source, like NTP, can be
obtained.
*May  3 15:14:55.456: %LINK-3-UPDOWN: Interface TenGigabitEthernet0/0/0, changed state to up
*May  3 15:14:55.458: %LINK-3-UPDOWN: Interface Vlan1, changed state to down
*May  3 15:14:55.456: %LINK-3-UPDOWN: SIP0/0: Interface TenGigabitEthernet0/0/0, changed state to up
*May  3 15:14:57.892: %LINEPROTO-5-UPDOWN: Line protocol on Interface TenGigabitEthernet0/0/0, changed state to up
*May  3 15:14:58.891: %LINK-3-UPDOWN: Interface Vlan1, changed state to up
*May  3 15:14:59.892: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to up
*May  3 15:15:09.367: %IOSXE_REDUNDANCY-6-PEER: Active detected chassis 2 as standby.
*May  3 15:15:09.365: %STACKMGR-6-STANDBY_ELECTED: Chassis 1 R0/0: stack_mgr: Chassis 2 has been elected STANDBY.
*May  3 15:15:09.652: %PMAN-3-PROC_EMPTY_EXEC_FILE: Chassis 2 R0/0: pvp: Empty executable used for process bt_logger
*May  3 15:15:10.140: %PMAN-3-PROC_EMPTY_EXEC_FILE: Chassis 2 R0/0: pvp: Empty executable used for process ngiolite
*May  3 15:15:14.751: %IOSXE_PEM-6-INSPEM_FM: PEM/FM slot P0 inserted
*May  3 15:15:14.754: %IOSXE_PEM-6-PEMOK: The PEM in slot P0 is functioning properly
*May  3 15:15:14.754: %IOSXE_PEM-6-INSPEM_FM: PEM/FM slot P2 inserted
*May  3 15:15:14.758: %IOSXE_PEM-6-PEMOK: The PEM in slot P2 is functioning properly
WLC>
```

```
WLC#
*May  3 15:15:39.434: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby insertion (raw-event=PEER_FOUND(4))

*May  3 15:15:39.434: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby insertion (raw-event=PEER_REDUNDANCY_STATE_CHANGE(5))

*May  3 15:15:41.404: % Redundancy mode change to SSO

*May  3 15:15:41.404: %VOICE_HA-7-STATUS: NONE->SSO; SSO mode will not take effect until after a platform reload.
*May  3 15:15:44.413: Syncing vlan database
*May  3 15:15:44.436: Vlan Database sync done from bootflash:vlan.dat to stby-bootflash:vlan.dat (1464 bytes)
WLC#
WLC#
WLC#
WLC#
WLC#
WLC#
WLC#show chas
WLC#show chassis
Chassis/Stack Mac Address : 00a3.8e23.8769 - Local Mac Address
Mac persistency wait time: Indefinite
Local Redundancy Port Type: Twisted Pair
                                           H/W   Current
Chassis#   Role    Mac Address     Priority Version  State                 IP
-------------------------------------------------------------------------------------
*1       Active   00a3.8e23.8769     1       V02      Ready                172.20.226.134
 2       Standby  00a3.8e23.8909     1       V02      HA sync in progress  172.20.226.133
```

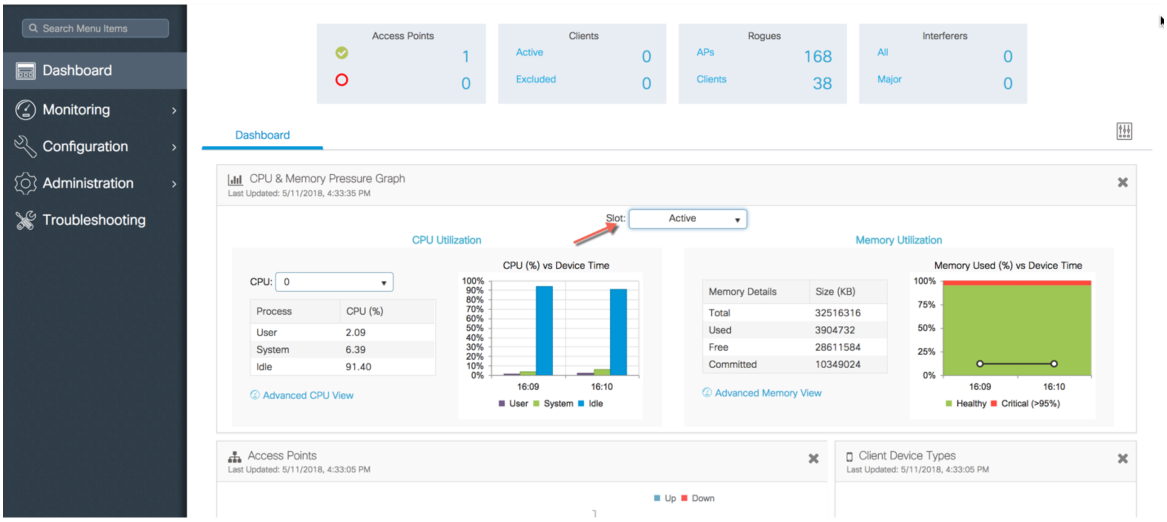
4. Terminal State for SSO

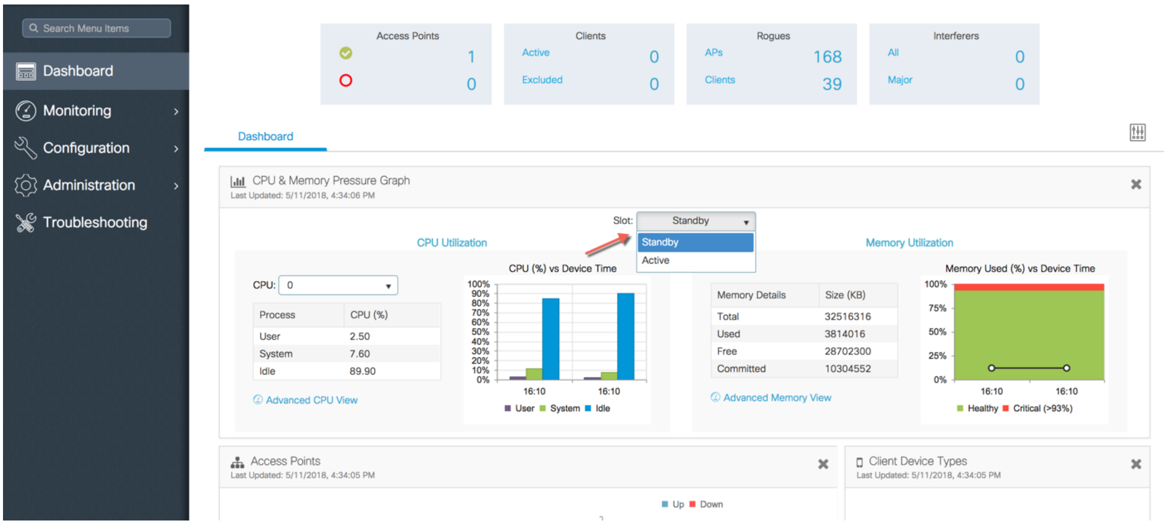
```
*May  3 15:18:46.564: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succee
*May  3 15:18:46.565: %VOICE_HA-7-STATUS: VOICE HA bulk sync done.
*May  3 15:18:47.565: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)
WLC#show chassis
Chassis/Stack Mac Address : 00a3.8e23.8769 - Local Mac Address
Mac persistency wait time: Indefinite
Local Redundancy Port Type: Twisted Pair
                                           H/W   Current
Chassis#   Role    Mac Address     Priority Version  State                 IP
-------------------------------------------------------------------------------------
*1       Active   [redacted]         1       V02      Ready                [redacted]
 2       Standby                     1       V02      Ready
```

Note: Breaking the HA Pair : The HA configuration can be disabled by using the chassis clear command followed by a reload

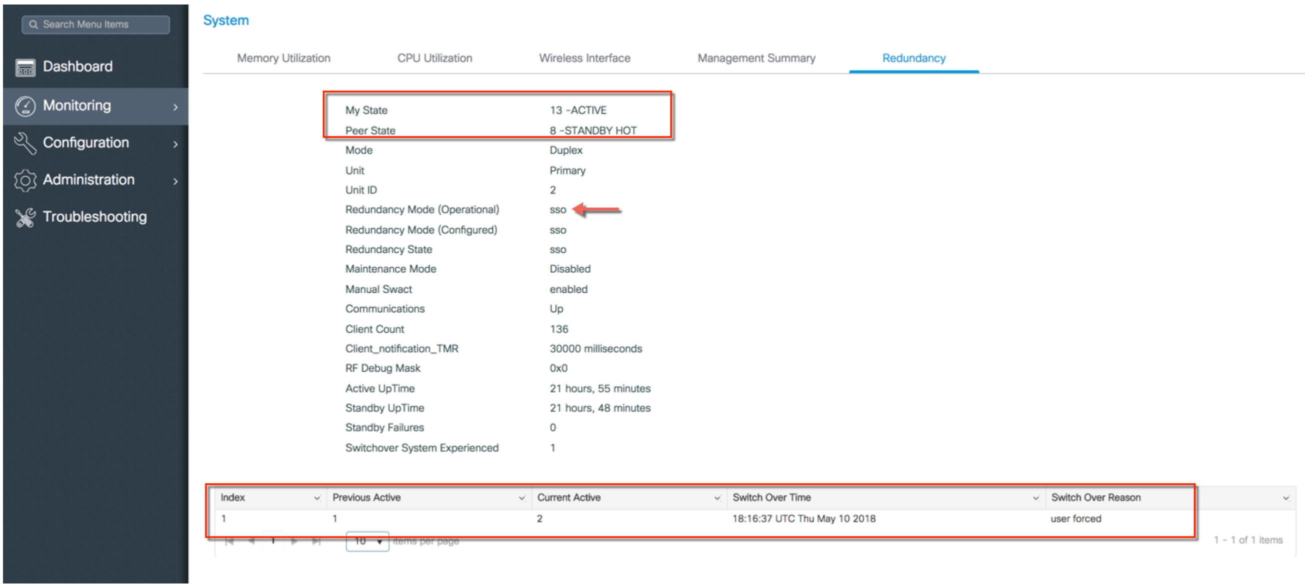
Monitoring the HA Pair

Both Active and Standby System can be monitored from the Management UI of the Active wireless controller. This includes information about CPU and memory utilization as well and advanced CPU and memory views.





Navigate to Monitoring > System > Redundancy on the controller Web UI. The Redundancy States page is displayed:



Parameter	**Description**
My State	Shows the state of the active CPU controller module. Values are as follows: Active Standby HOT Disable

Peer State	Displays the state of the peer (or standby) CPU controller module. Values are as follows: Standby HOT Disable
Mode	Displays the current state of the redundancy peer. Values are as follows: Simplex– Single CPU controller module. Duplex– Two CPU controller modules.
Unit ID	Displays the unit ID of the CPU controller module.
Redundancy Mode (Operational)	Displays the current operational redundancy mode supported on the unit.
Redundancy Mode (Configured)	Displays the current configured redundancy mode supported on the unit.
Redundancy State	Displays the current functioning redundancy state of the unit. Values are as follows: SSO Not Redundant
Manual Swact	Displays whether manual switchovers have been enabled.
Communications	Displays whether communications are up or down between the two controllers.

The same page displays Switchover history. The description for the following parameters are displayed in the table below:

Parameter	**Description**
Index	Displays the index number of the redundant unit.
Previous Active	Displays the controller that was active prior to switchover.

Current Active	Displays the controller that is currently active.
Switch Over Time	Displays the system time when the switchover occurred.
Switch Over Reason	Displays the cause of the switchover.

Monitoring HA Pair from CLI

The command `show chassis` displays summary information about the HA Pair, including the MAC address, role, switch priority, and current state of each wireless controller in the redundant HA pair. By default, the Local MAC Address of the HA Pair is the MAC address of the first elected Active Controller.

```
WLC#show chassis
Chassis/Stack Mac Address : 00a3.8e23.8760 - Local Mac Address
Mac persistency wait time: Indefinite
Local Redundancy Port Type: Twisted Pair
                                             H/W   Current
Chassis#   Role     Mac Address     Priority Version  State                  IP
-------------------------------------------------------------------------------------
 1        Standby  00a3.8e23.8760      1       V02     Ready              172.20.226.133
*2        Active   00a3.8e23.8900      1       V02     Ready              172.20.226.134
```

The `show chassis` command points to the current C9800 wireless controller on the console using the (*) symbol against the chassis number as shown above.

Verifying Redundancy States

■ The command show redundancy can be used to monitor the state of the two units

```
wireless controller#show redundancy ?
  application       box 2 box application information
  clients           Redundancy Facility (RF) client list
  config-sync       Show Redundancy Config Sync status
  counters          Redundancy Facility (RF) operational counters
  domain            Specify the RF domain
  history           Redundancy Facility (RF) history
  idb-sync-history  Redundancy Facility (RF) IDB sync history
  linecard-group    Line card redundancy group information
  rii               Display the redundancy interface identifier for Box to Box
  states            Redundancy Facility (RF) states
  switchover        Redundancy Facility (RF) switchover
  trace             Redundancy Facility (RF) trace
  |                 Output modifiers
  <cr>              <cr>
```

■ The command show redundancy displays the redundant system and the current processor information. The redundant system information includes the system uptime, standby failures, switchover reason, hardware mode, and configured and operating redundancy mode. The current processor information displayed includes the image version, active location, software state, BOOT variable, configuration register value, and uptime in the current state, and so on. The Peer Processor information is only available from the Active Controller.

```
WLC#show redundancy
Redundant System Information :
------------------------------
       Available system uptime = 22 hours, 9 minutes
Switchovers system experienced = 1
              Standby failures = 0
        Last switchover reason = user forced

                 Hardware Mode = Duplex
    Configured Redundancy Mode = sso
     Operating Redundancy Mode = sso
              Maintenance Mode = Disabled
                Communications = Up

Current Processor Information :
-------------------------------
               Active Location = slot 2
        Current Software state = ACTIVE
       Uptime in current state = 21 hours, 43 minutes
                 Image Version = Cisco IOS Software [Fuji], WLC9000 Software (X86_64_LINUX_IO
SD-UNIVERSALK9_WLC-M), Experimental Version 16.10.20180509:065558 [polaris_dev-/nobackup/mcpr
e/BLD-BLD_POLARIS_DEV_LATEST_20180509_073715 183]
Copyright (c) 1986-2018 by Cisco Systems, Inc.
Compiled Wed 09-May-18 06:35 by mcpre
                          BOOT = bootflash:qwlc-universalk9_wlc.BLD_POLARIS_DEV_LATEST_201805
09_073715.SSA.bin,1;
                   CONFIG_FILE =
        Configuration register = 0x2102

Peer Processor Information :
----------------------------
              Standby Location = slot 1
        Current Software state = STANDBY HOT
       Uptime in current state = 21 hours, 35 minutes
                 Image Version = Cisco IOS Software [Fuji], WLC9000 Software (X86_64_LINUX_IO
SD-UNIVERSALK9_WLC-M), Experimental Version 16.10.20180509:065558 [polaris_dev-/nobackup/mcpr
e/BLD-BLD_POLARIS_DEV_LATEST_20180509_073715 183]
Copyright (c) 1986-2018 by Cisco Systems, Inc.
Compiled Wed 09-May-18 06:35 by mcpre
                          BOOT = bootflash:qwlc-universalk9_wlc.BLD_POLARIS_DEV_LATEST_201805
09_073715.SSA.bin,1;
                   CONFIG_FILE =
        Configuration register = 0x2102
```

- The command show redundancy states displays all the redundancy states of the active and standby controllers.

```
WLC#show redundancy states ?
  domain  Specify the RF domain
  |       Output modifiers
  <cr>    <cr>

WLC#show redundancy states
       my state = 13 -ACTIVE
     peer state = 8  -STANDBY HOT
           Mode = Duplex
           Unit = Primary
        Unit ID = 2

Redundancy Mode (Operational) = sso
Redundancy Mode (Configured)  = sso
Redundancy State              = sso
     Maintenance Mode = Disabled
    Manual Swact = enabled
 Communications = Up

   client count = 136
 client_notification_TMR = 30000 milliseconds
          RF debug mask = 0x0
```

- Manual Switchover Action (Manual Swact) i.e. the command redundancy force-switchover cannot be executed on the Standby wireless controller and is enabled only on the Active Controller.

- Switchover History can be viewed using the following command

```
WLC#show redundancy switchover history
Index  Previous  Current  Switchover             Switchover
       active    active   reason                 time
-----  --------- -------- ----------             ----------
   1       1        2     user forced            18:16:37 UTC Thu May 10 2018
```

Accessing standby wireless controller console

The active controller can be accessed through a console connection, Telnet, an SSH, or a Web Browser by using the Management IP address. To use the console on the standby wireless controller, execute the following commands from the active Catalyst 9800 Wireless controller

```
conf t
redundancy
main-cpu
standby console enable
```

The prompt on the Standby console is appended with "-stby" to reflect the Standby wireless controller console as shown below.

```
WLC-stby#show chassis
Chassis/Stack Mac Address : 00a3.8e23.8760 - Local Mac Address
Mac persistency wait time: Indefinite
Local Redundancy Port Type: Twisted Pair
                                             H/W   Current
Chassis#   Role     Mac Address      Priority Version  State                 IP
------------------------------------------------------------------------------------
*1       Standby  00a3.8e23.8760     1       V02     Ready                0.0.0.0
 2       Active   00a3.8e23.8900     1       V02     Ready                0.0.0.0
```

Note: The `show chassis` command points to the current C9800 wireless controller on the console using the (*) symbol against the chassis number as shown above. In this case it is the console of the standby Unit.

```
WLC-stby>en
WLC-stby#show red
WLC-stby#show redun
WLC-stby#show redundancy
Redundant System Information :
------------------------------
       Available system uptime = 22 hours, 2 minutes
Switchovers system experienced = 1

                 Hardware Mode = Duplex
    Configured Redundancy Mode = sso
     Operating Redundancy Mode = sso
              Maintenance Mode = Disabled
                Communications = Up

Current Processor Information :
-------------------------------
              Standby Location = slot 1
        Current Software state = STANDBY HOT
       Uptime in current state = 21 hours, 29 minutes
                 Image Version = Cisco IOS Software [Fuji], WLC9000 Software (X86_64_LINUX_IO
SD-UNIVERSALK9_WLC-M), Experimental Version 16.10.20180509:065558 [polaris_dev-/nobackup/mcpr
e/BLD-BLD_POLARIS_DEV_LATEST_20180509_073715 183]
Copyright (c) 1986-2018 by Cisco Systems, Inc.
Compiled Wed 09-May-18 06:35 by mcpre
                          BOOT = bootflash:qwlc-universalk9_wlc.BLD_POLARIS_DEV_LATEST_201805
09_073715.SSA.bin,1;
                   CONFIG_FILE =
        Configuration register = 0x2102

Peer (slot: 2, state: ACTIVE) information is not available because this is the standby proces
sor
```

Switchover Functionality

Process Failure Switchover

This type of switch over occurs when any of the key processes running on the Active unit fails or crashes. Upon such a failure, the Active unit reloads and the hot Standby takes over and becomes the new Active unit. When the failed system boots up, it will transition to Hot-Standby state. If the Standby unit is not yet in Hot Standby State, both units are reloaded and there will be no SSO. A process failure on the standby (hot or not) will cause it to reload.

Power-fail Switchover

This switchover from the Active to Standby unit is caused due to power failure of the current Active unit. The current Standby unit becomes the new Active unit and when the failed system boots up, it will transition to Hot-Standby state.

Manual Switchover

This is a user initiated forced switchover between the Active and Standby unit. The current Standby unit becomes the new Active unit and when the failed system boots up, it will transition to Hot-Standby state. To perform a manual switchover, execute the redundancy force-switchover command. This command initiates a graceful switchover from the active to the standby controller. The active controller reloads and the standby takes over as the New Active controller.

Failover Process

Active wireless controller

```
WLC#show ap summary
Number of APs: 1

AP Name                                  Slots    AP Model  Ethernet MAC    Radio MAC       Location             Country     IP Address
State
--------------------------------------------------------------------------------------------------------------------------------------------------
----------------------------------------
AP005D.735C.B544                           3      3802I     005d.735c.b544  b4de.31d0.5800  default location     US          172.20.226.186
Registered


WLC#show wireless client sum
Number of Local Clients: 1

MAC Address    AP Name                          WLAN  State              Protocol Method     Role
-------------------------------------------------------------------------------------------------------
e8b2.ac94.757e AP005D.735C.B544                 1     Run                11ac     None       Local

Number of Excluded Clients: 0

WLC#redundancy force-switchover

System configuration has been modified. Save? [yes/no]: yes
Building configuration...
[OK]Proceed with switchover to standby RP? [confirm]
    Manual Swact = enabled

Chassis 1 reloading, reason - Non participant detected
```

Standby wireless controller

An Access Point and client Stateful Switch Over (SSO) implies that all the Access Point and client sessions are switched over state-fully and continue to operate in a network with no loss of sessions, providing improved network availability and reducing service downtime.

Once a redundancy pair is formed, HA is enabled, which means that Access Points and clients continue to remain connected during an active-to-standby switchover.

```
WLC-stby#
May 10 18:16:37.123: %PLATFORM-6-HASTATUS: RP switchover, received chassis event to become active
May 10 18:16:37.169: %REDUNDANCY-3-SWITCHOVER: RP switchover (PEER_NOT_PRESENT)
May 10 18:16:37.169: %REDUNDANCY-3-REDUNDANCY_ALARMS: Unable to assert REDUNDANCY alarm

May 10 18:16:37.169: %REDUNDANCY-3-REDUNDANCY_ALARMS: Unable to assert REDUNDANCY alarm

May 10 18:16:37.169: %REDUNDANCY-3-SWITCHOVER: RP switchover (PEER_DOWN)
May 10 18:16:37.169: %REDUNDANCY-3-SWITCHOVER: RP switchover (PEER_REDUNDANCY_STATE_CHANGE)
May 10 18:16:37.175: %PLATFORM-6-HASTATUS: RP switchover, sent message became active. IOS is ready to switch to primary after chassis
confirmation
May 10 18:16:37.180: %PLATFORM-6-HASTATUS: RP switchover, received chassis event became active
May 10 18:16:37.789: %VOICE_HA-2-SWITCHOVER_IND: SWITCHOVER, from STANDBY_HOT to ACTIVE state.
May 10 18:16:37.797: %LINK-3-UPDOWN: Interface Lsmpi0, changed state to up
May 10 18:16:37.798: %LINK-3-UPDOWN: Interface EOBC0, changed state to up
May 10 18:16:37.798: %LINK-3-UPDOWN: Interface LIIN0, changed state to up
May 10 18:16:38.798: %LINEPROTO-5-UPDOWN: Line protocol on Interface Lsmpi0, changed state to up
May 10 18:16:38.798: %LINEPROTO-5-UPDOWN: Line protocol on Interface EOBC0, changed state to up
May 10 18:16:38.798: %LINEPROTO-5-UPDOWN: Line protocol on Interface LIIN0, changed state to up
May 10 18:16:39.786: %LINK-3-UPDOWN: Interface Null0, changed state to up
May 10 18:16:39.786: %LINK-3-UPDOWN: Interface TenGigabitEthernet0/0/0, changed state to up
May 10 18:16:39.787: %LINK-3-UPDOWN: Interface Vlan1, changed state to up
May 10 18:16:39.788: %LINK-3-UPDOWN: Interface Vlan112, changed state to up
May 10 18:16:40.787: %LINEPROTO-5-UPDOWN: Line protocol on Interface Null0, changed state to up
May 10 18:16:40.787: %LINEPROTO-5-UPDOWN: Line protocol on Interface TenGigabitEthernet0/0/0, changed state to up
May 10 18:16:40.787: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to up
May 10 18:16:40.787: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan112, changed state to up
WLC#
May 10 18:16:49.798: %LINK-3-UPDOWN: Interface GigabitEthernet0, changed state to up
May 10 18:16:50.799: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0, changed state to up
WLC#show ap sum
WLC#show ap summary
Number of APs: 1

AP Name                                  Slots    AP Model  Ethernet MAC    Radio MAC       Location             Country     IP Address
State
--------------------------------------------------------------------------------------------------------------------------------------------------
----------------------------------------
AP005D.735C.B544                           3      3802I     005d.735c.b544  b4de.31d0.5800  default location     US          172.20.226.186
Registered

WLC#show wireless client summary
Number of Local Clients: 1

MAC Address    AP Name                          WLAN  State              Protocol Method     Role
-------------------------------------------------------------------------------------------------------
e8b2.ac94.757e AP005D.735C.B544                 1     Run                11ac     None       Local

Number of Excluded Clients: 0
```

Verifying AP and Client SSO State Sync

On successful switchover of the standby wireless controller as active, all access points and clients connected to the previously active wireless controller must remain connected to the new Active controller.

This can be verified by executing the commands:

- **show ap uptime** : Verifies that the uptime of the access point after the switchover is not reset.
- **show wireless client summary**: Displays the clients connected to the new Active controller.

```
WLC#show ap uptime
Number of APs: 1

AP Name                          Ethernet MAC    Radio MAC       AP Up Time                                              Association Up Time
------------------------------------------------------------------------------------------------------------------------------------------------
--
AP005D.735C.B544                 005d.735c.b544  b4de.31d0.5800  1 day 0 hour 47 minutes 22 seconds                      1 day 0 hour 45 minutes 33 s
econds

WLC#
```

```
WLC#show wireless client summary
Number of Local Clients: 1

MAC Address    AP Name                          WLAN  State            Protocol Method    Role
-------------------------------------------------------------------------------------------------
e8b2.ac94.757e AP005D.735C.B544                 1     Run              11ac     None      Local

Number of Excluded Clients: 0
```

SSO Failover Time Metrics

Metrics	Time
Failure Detection	In the order of 500-1000ms

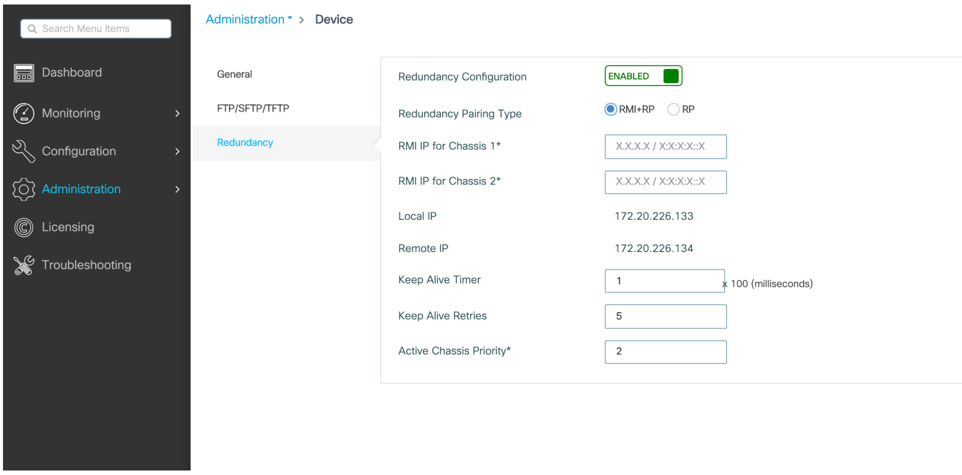
Redundancy Management Interface

With a single RP link between the SSO pair, if the heartbeat on RP fails, there is no way find out if the failure is limited to the link or if the other controller has failed. Redundancy Port (RP link) that handles state sync traffic between the active and the standby is a single point of failure.

Release 17.1 introduces the Redundancy Management Interface (RMI) as a secondary link between the active and the standby controllers. This release also introduces the support for default gateway check which is done using the redundancy management interface.

Release 17.3.2 introduces IPv6 Support for RMI interfaces. One management IPv6 address and one RMI IPv6 address is supported on the wireless management interface. Either RMI IPv4 or RMI IPv6 is supported and there is no simultaneous support for RMI IPv4 and RMI IPv6. The format of the CLI is same for IPv6 except that the IPv4 address is replaced with IPv6 address.

Redundancy Management Interface Configuration using WebUI



- RMI IP for chassis 1 and 2 is same across both active and standby controllers
- RP IP configuration for chassis 1 and 2 auto-generated as 169.254.x.x where x.x. is from the RMI IP
- The netmask for RMI is picked up from the netmask configured on the Wireless Management VLAN.
- WebUI has RMI IPv6 support in Release 17.3.2

Programmatic configuration of RMI IPs

On the Active controller:

Secondary address on the management VLAN is the RMI for the active. The primary address on the active is the management IP. It is possible to have multiple “secondary” addresses on the interface as shown below. For the purpose of RMI, only one secondary IP will be defined. The secondary IP shall be configured programmatically.

There is no concept of “secondary” address in case of IPv6. The wireless management IP and the RMI IP will appear as 2 distinct IPs in case of IPv6.

For eg, if the following CLI is configured: redun-management interface Vlan52 chassis 1 address 2020:0:0:1::211 chassis 2 address 2020:0:0:1::212

The active controller will be configured as follows:

```
interface Vlan52
 ip address 10.100.0.1 255.252.0.0
 ipv6 address 2020:0:0:1::1/64
 ipv6 address 2020:0:0:1::211/64
 ipv6 enable
 ipv6 nd na glean
 no mop enabled
 no mop sysid
end
```

On the Standby controller:

It cannot have the management IP as the address is claimed by the active. Therefore, on the standby controller, the RMI IP shall be configured as the primary address programmatically. When the standby becomes active, the management IP needs to be programmed as primary and the RMI IP as secondary.

The “secondary” IP concept is relevant for IPv4 only.

```
interface Vlan52
 no ip address
 ipv6 address 2020:0:0:1::212/64
 ipv6 enable
 ipv6 nd na glean
 no mop enabled
 no mop sysid
end
```

Dual Stack support with RMI IPv4

When RMI IPv4 is configured, it is possible to an IPv6 IP configured on the wireless management interface. This address shall be explicitly configured. With RMI enabled, the IPv6 address configured shall be programmatically removed in the standby and configured back when the standby transitions to active. The address shall be removed when the controller is in active-recovery mode. This would avoid Duplicate Address Detection.

Dual Stack Support with RMI IPv6

This case arises in release 17.3.2. In 17.3.2, the wireless management IP can be IPv6 with an RMI IPv6 configured. In addition, the wireless management interface can have an IPv4 IP configured. When the standby RMI interface is brought UP, the IPv6 and IPv4 management IPs will be unconfigured and IPv6 RMI configured. Upon transition from standby to active, the management IPs shall be restored.

Peer Timeout Configuration

- Active and standby chassis send keepalives messages to each other to ensure both still available. Peer timeout is used to determine peer chassis is lost if it does not receive any keep alive message from peer chassis in the configured peer timeout.

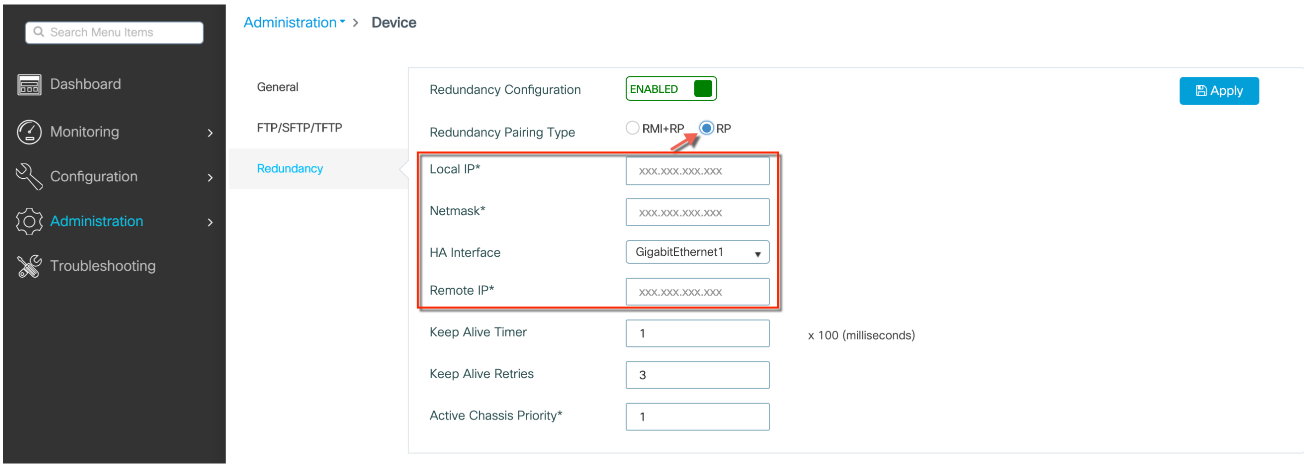
- Default timeout is 100ms but is configurable up to 1000 ms. The keepalive retries are 5 by default but can be configured all the way to 10.

- CLI commands:

```
WLC#chassis redundancy keep-alive timer ?

        <1-10>  Chassis peer keep-alive time interval in multiple of 100 ms (enter 1
        for default)
WLC#chassis redundancy keep-alive retries ?
        <5-10>  Chassis peer keep-alive retries before claiming peer is down (enter 5
for default)
```

For backward compatibility, RP based SSO configuration will also be supported, but keep in mind that this will not support default gateway check and hence is not preferred.



Redundancy Management Interface Configuration using CLI

Until 17.1, only RP-based SSO configuration was supported, i.e., chassis redundancy ha-interface <RP interface> local-ip <local IP> <local IP subnet> remote-ip <remote IP>.

17.1 and beyond, the user can use either RMI+RP or RP-based configuration. Once an HA pair is formed using RMI+RP configuration, the exec CLI for RP-based method of clearing and forming the HA pair shall not be allowed.

Note: Chassis re-number needs to be configured while bringing up HA with RMI from scratch using RMI in 17.x release.

By default, chassis number is 1. IP addresses of RP ports are derived from RMI. If the chassis number is the same on both controllers, local RP port IP derivation will be same and discovery will fail. This will result in Active-Active case.

To avoid this scenario, execute the following CLI:

```
WLC#chassis 1 renumber ?
  <1-2>  Renumber local chassis id assignment

WLC(config)# redun-management interface <VLAN> chassis 1 address <RMI IP of chassis 1>
chassis 2 address <RMI IP of chassis 2>
```

Configuration example:

On WLC 1:

```
WLC(config)# redun-management interface Vlan112 chassis 1 address 172.20.226.148 chassis 2
address 172.20.226.149
```

On WLC 2: (Same CLI)

```
WLC(config)# redun-management interface Vlan112 chassis 1 address 172.20.226.148 chassis 2
address 172.20.226.149
```

Chassis numbers identify the individual controllers and must be configured before configuring the RMI IPs. It is mandatory to execute the same CLI on both controllers before forming the pair. The RMI IP configuration triggers HA pairing and forms the SSO pair.

Verifying RMI and RP configuration

```
WLC-9800#show chassis rmi
Sep 20 21:26:13.024: %SYS-5-CONFIG_I: Configured from console by console
Chassis/Stack Mac Address : 00a3.8e23.8760 - Local Mac Address
Mac persistency wait time: Indefinite
Local Redundancy Port Type: Twisted Pair
                                             H/W   Current
Chassis#   Role     Mac Address     Priority Version  State       IP           RMI-IP
------------------------------------------------------------------------------------
 1        Standby  00a3.8e23.8760     2       V02      Ready  169.254.226.149 172.20.226.149
*2        Active   00a3.8e23.8900     1       V02      Ready  169.254.226.148 172.20.226.148


WLC-9800#show romvar
ROMMON variables:
 SWITCH_NUMBER = 1
 LICENSE_BOOT_LEVEL =
…
 RANDOM_NUM = 842430634
 SWITCH_PRIORITY = 1
 RMI_INTERFACE_NAME = Vlan112
 RMI_CHASSIS_LOCAL_IP = 172.20.226.148
 RMI_CHASSIS_REMOTE_IP = 172.20.226.149
 CHASSIS_HA_LOCAL_IP = 169.254.226.148
 CHASSIS_HA_REMOTE_IP = 169.254.226.149
 CHASSIS_HA_LOCAL_MASK = 255.255.255.0


The following shows the scenario where the RP IP is derived from RMI IPv6 address:
```

```
D3-5-Dao#show chassis rmi
Chassis/Stack Mac Address : 00a3.8e23.a540 - Local Mac Address
Mac persistency wait time: Indefinite
Local Redundancy Port Type: Twisted Pair
                                        H/W   Current
Chassis#   Role     Mac Address     Priority Version  State               IP                RMI-IP
------------------------------------------------------------------------------------------------------
*1       Active   706d.1536.23c0     1       V02      Ready            169.254.254.17    2020:0:0:1::211
 2       Standby  00a3.8e23.a540     1       V02      Ready            169.254.254.18    2020:0:0:1::212
```

RMI and RP pairing combinations

Upgrade and HA Pairing with no previous HA config

The user shall be presented with an option to choose the existing mechanism (exec RP-based CLIs) or the RMI IP based mechanism.

If the user chooses the exec CLI based method, the RP IPs shall be configured as it happens till 16.12.

When the RMI configuration is done, it shall:

Generate the RP IPs with IPs derived from the RMI IPs and will also be used for setting RMI IPs and pair the

Controllers (while pairing only standby reloads in hardware platforms. Both active and standby reload in case of 9800-CL VM). Exec RP-based CLIs are blocked in this case.

Option 1: RMI Based Configuration (Preferred)

1. Upgrade to 17.1 and connect the RPs
2. Configure RMI+RP
3. RP IPs are derived from the RMI IPs
4. RP-based exec commands are blocked
5. ROMMON RP and RMI variables are set

Option 2: RP Based Configuration

1. Upgrade to 17.1 and connect RPs
2. Configure RP via GUI/CLI
3. RP-based configuration sets the local and remote IP
4. ROMMON RP Variables are set to the local and remote IP

Upgrade already Paired controllers

If the controllers are already in an HA pair, the existing exec RP CLIs can be continued to be used.

Those who would like to migrate to the RMI based HA pairing (preferred) can enable RMI.

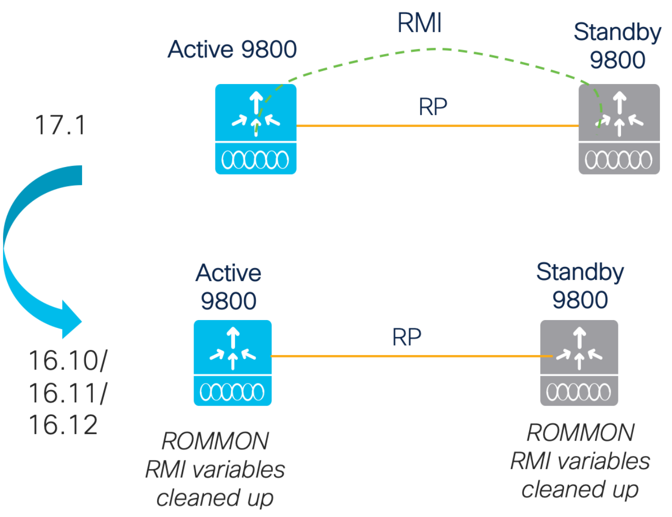
This will overwrite the RP IPs with RMI derived IPs. The HA pair will not be immediately disturbed, but the controllers will pick up the new IP when they reload next.

RMI feature mandates a reload for the feature to take effect.

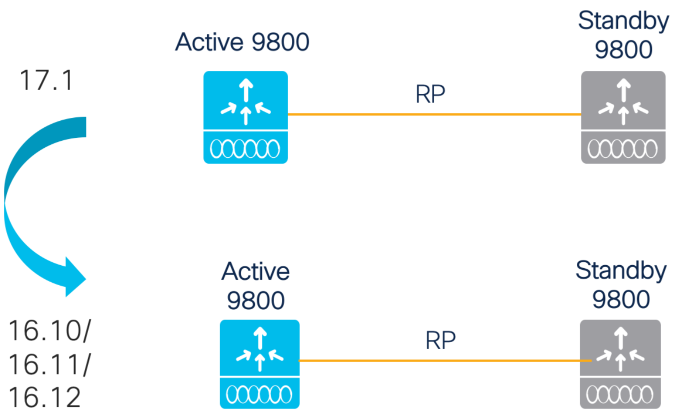
When the controllers reload, they would come up as a pair with the new RMI-derived-RP-IPs. Exec RP-based CLIs will be blocked

Downgrade

If RMI based configuration was used, after downgrade the system will fall back to the RP-based configuration



If RP based configuration was used, after downgrade the system will continue to use RP-based configuration

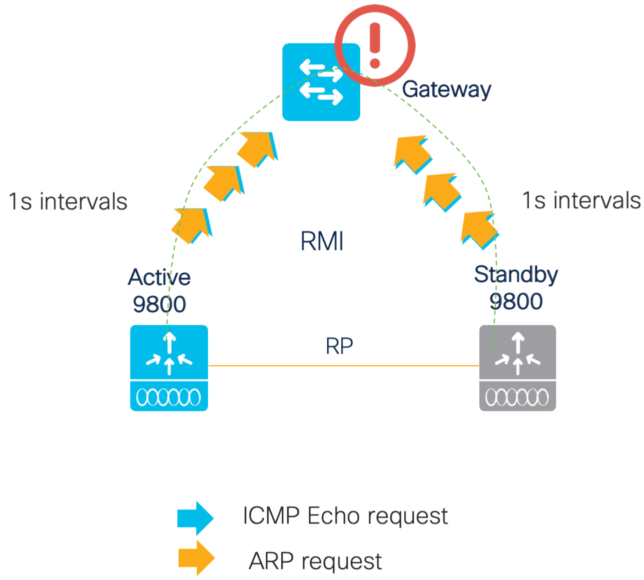


Default Gateway Check

Default Gateway check is done by periodically sending Internet Control Message Protocol (ICMP) ping to the gateway. Both the active and the standby controllers use the RMI IP as the source IP. These messages are sent at 1 second interval. If there are 8 consecutive failures in reaching the gateway, the controller will declare the gateway as non-reachable.

After 4 ICMP Echo requests fail to get ICMP Echo responses, ARP requests are attempted. If there is no response for 8 seconds (4 ICMP Echo Requests followed by 4 ARP Requests), the gateway is assumed to be non-reachable.

IPv6 default gateway detection is supported starting release 17.3.2. Instead of ICMP and ARP in IPv4, IPv6 shall use ICMP to detect gateway failure.



The Catalyst 9800 Wireless controller has two recovery states to prevent an active-active scenario.

Recovery mode logically means a state where the controller does not have all “resources” available to provide the service. Currently, RP, RMI and Gateway are the resources. Ports will be in admin down in recovery mode, so no traffic goes through.

- Standby-Recovery: If Gateway goes down, standby goes to standby-recovery mode. Standby means, its state is up to date with the active. But since it does not have the other resource (Gateway) it goes to Standby-Recovery. The standby shall not be in a position to take over the active functionality when it is in standby-recovery mode. Standby-Recovery will go back to Standby without a reload, once it detects that the Gateway reachability is restored.

- Active-Recovery is when the RP goes down. Active-Recovery does not have its internal state in sync with the Active. Active-Recovery will reload when the RP link comes up so that it can come up as Standby with bulk sync.

Switchover history will show switchover reason as Gateway down in the event of a switchover triggered as a result of the gateway going down.

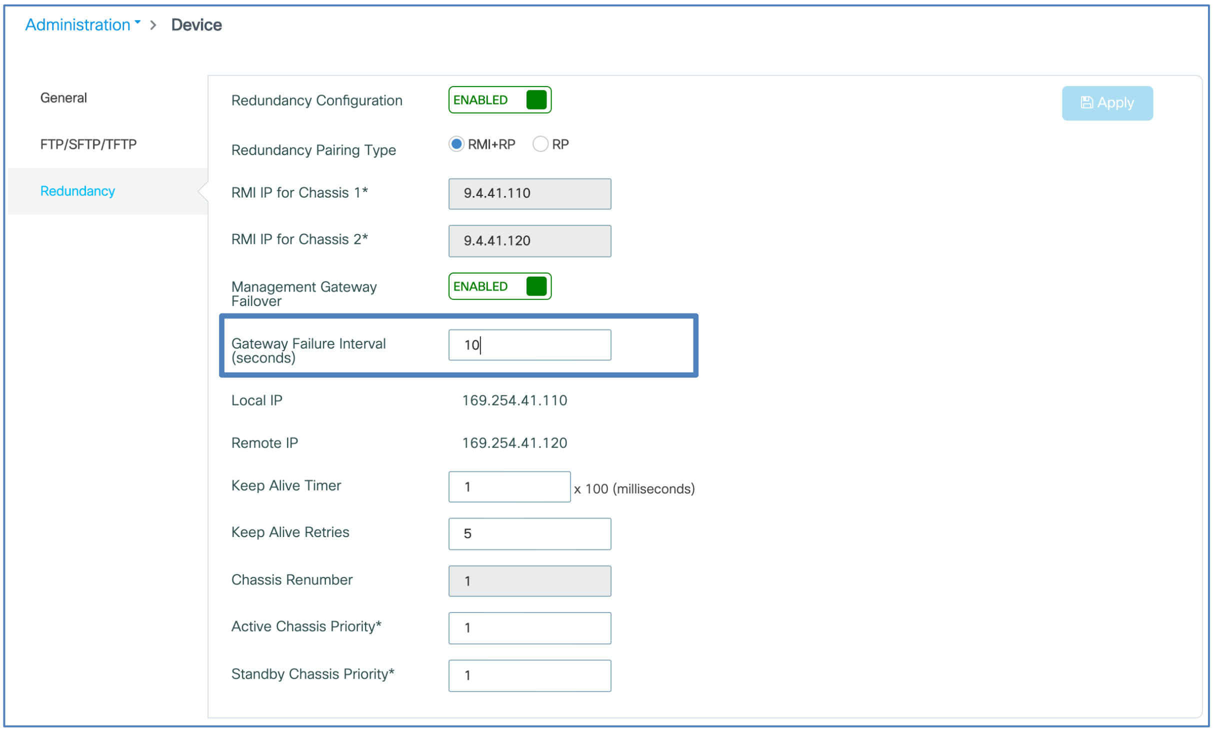
Configuring Gateway Failure Detection Interval

The gateway failure detection interval is configurable starting release 17.3.2 using the following CLI:

```
WLC(config)#management gateway-failover interval <6 - 12>
```

The default is 8.

This parameter can be configured through YANG, SNMP and WebUI as well. The configuration parameter is applicable for IPv6 gateway monitoring also.

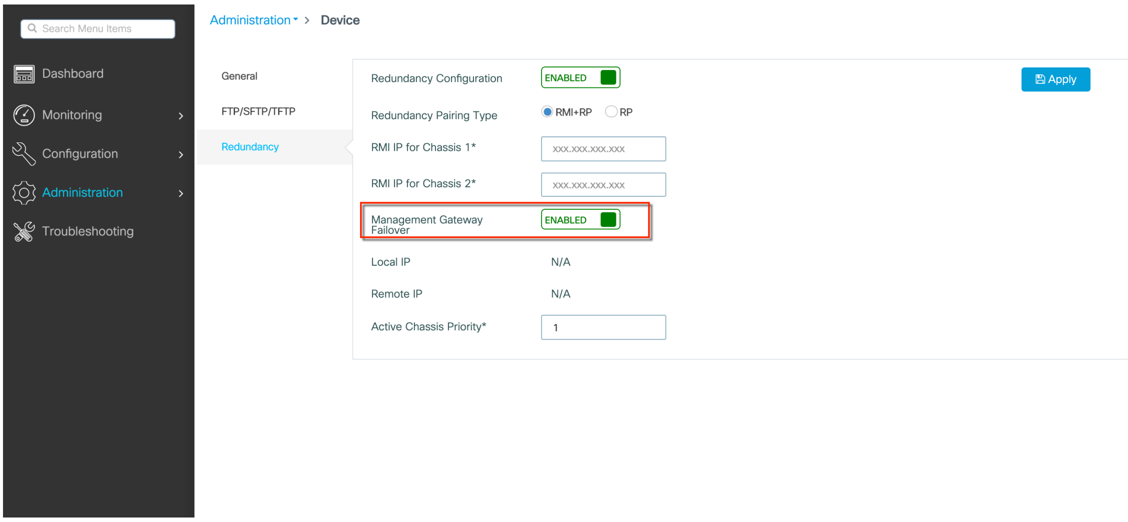


Sample json for NETCONF/YANG support


```
{
    "Cisco-IOS-XE-native:management": {
        "Cisco-IOS-XE-rmi-dad:gateway-failover": {
            "enable": true,
            "interval": 10
        }
    }
}
```

Default Gateway Check WebUI Configuration

The default gateway check option can be configured under Administration > Device > Redundancy > Management Gateway Failover



Default Gateway Check CLI Configuration

The following CLIs need to be configured for the gateway check functionality to be enabled and to specify the default gateway IP used by this feature

```
WLC-9800(config)#management gateway-failover enable
WLC-9800#ip default-gateway <IP>
```

To verify if gateway check is enabled, use the CLI show redundancy state

```
WLC-9800#show redundancy states
       my state = 13 -ACTIVE
     peer state = 8  -STANDBY HOT
           Mode = Duplex
           Unit = Primary
        Unit ID = 2
Redundancy Mode (Operational) = sso
Redundancy Mode (Configured)  = sso
Redundancy State              = sso
…
Gateway Monitoring = Enabled
```

With 17.2, usage of “ip default-gateway <IP>” shall be removed . Gateway IP will be picked up from the static IP routes configured. The HA infrastructure will choose the static route IP that matches the RMI network. If there are multiple static routes configured, the route configured for the broadest network scope shall be selected. It is possible to configure multiple gateways for the same network scope. If there are multiple gateways for the same network, broadest mask and least gateway IP is chosen. The gateway IP shall be reevaluated, if necessary, when config update to static routes happens.

The above mechanism of selecting the gateway IP from the set of static routes is applicable to IPv6 in Release 17.3.2.

Note:

•Physical port down scenario takes 8 seconds to be detected as it is detected via GW check mechanism prior to release 17.3.2. Starting release 17.3.2, if the port state goes down all actions associated with gateway going down will be triggered. A new reason code will be used to indicate SSO due to detection of port going down. “Active RMI Port Down” shall be used in place of “Active GW Lost”.

•Physical port status is synced from the active to standby controller in release 17.1. This has been fixed in release 17.2 and the active and standby controllers maintain their own port status.

System and Network Fault Handling

If the standby controller crashes, it shall reboot and come up as standby. Bulk sync will follow and the standby will become hot. If the active controller crashes, the standby becomes active. The new active shall assume the role of master and try to detect a dual active.

These matrices provide a clear picture of what condition the WLC Switchover will trigger:

System Issues				
Trigger	**RP Link Status**	**Peer Reachability through RMI**	**Switchover**	**Result**
Critical Process crash	Up	Reachable	Yes	Switchover happens
Forced switchover	Up	Reachable	Yes	Switchover happens
Critical Process crash	Up	Unreachable	Yes	Switchover happens
Forced switchover	Up	Unreachable	Yes	Switchover happens
Critical Process crash	Down	Reachable	No	No action, one controller will be in recovery mode already.
Forced switchover	Down	Reachable	N/A	No action, one controller will be in recovery mode already.
Critical Process crash	Down	Unreachable	No	Double fault – as mentioned in Network Error handling
Forced switchover	Down	Unreachable	N/A	Double fault – as mentioned in Network Error handling

RP Link	Peer reachability through RMI	Gateway From Active	Gateway from Standby	Switchover	Result
Up	Up	Reachable	Reachable	No	No action
Up	Up	Reachable	Unreachable	No	No Action. Standby is not ready for SSO in this state as it does not have gateway reachability. The standby shall be shown to be in standby-recovery mode. If the RP goes down, standby (in recovery mode) shall become active.
Up	Up	Unreachable	Reachable	Yes	Gateway reachability message is exchanged over the RMI + RP links. Active shall reboot so that standby becomes active.
Up	Up	Unreachable	Unreachable	No	With this, when the active SVI goes down, so will the standby SVI. A switchover is then triggered. If the new active discovers its gateway to be reachable, the system shall stabilize in Active - Standby Recovery.

					Otherwise, switchovers will happen in a ping-pong fashion.
Up	Down	Reachable	Reachable	No	No Action
Up	Down	Reachable	Unreachable	No	Standby is not ready for SSO in this state as it does not have gateway reachability. Standby will go to recovery mode as LMP messages are exchanged over the RP link also.
Up	Down	Unreachable	Reachable	Yes	Gateway reachability message is exchanged over RP link also. Active shall reboot so that standby becomes active.
Up	Down	Unreachable	Unreachable	No	With this, when the active SVI goes down, so will the standby SVI. A switchover is then triggered. If the new active discovers its gateway to be reachable, the system shall stabilise in Active - Standby Recovery. Otherwise,

					switchovers will happen in a ping-pong fashion.
Down	Up	Reachable	Reachable	Yes	Standby will become active with (old) active going to active-recovery. Config mode is disabled in active-recovery mode. All interfaces will be ADMIN DOWN with the wireless management interface having RMI IP. The controller in Active Recovery will reload to become standby when the RP link comes UP.
Down	Up	Reachable	Unreachable	Yes	Same as above
Down	Up	Unreachable	Reachable	Yes	Same as above
Down	Up	Unreachable	Unreachable	Yes	Same as above
Down	Down	Reachable	Reachable	Yes	Double fault – this may result in a network conflict as there will be 2 active controllers.

					Standby becomes active. Old active also exists. Role negotiation has to happen once the connectivity is restored and keep the active that came up last
Down	Down	Reachable	Unreachable	Yes	Same as above
Down	Down	Unreachable	Reachable	Yes	Same as Above
Down	Down	Unreachable	Unreachable	Yes	Same as Above

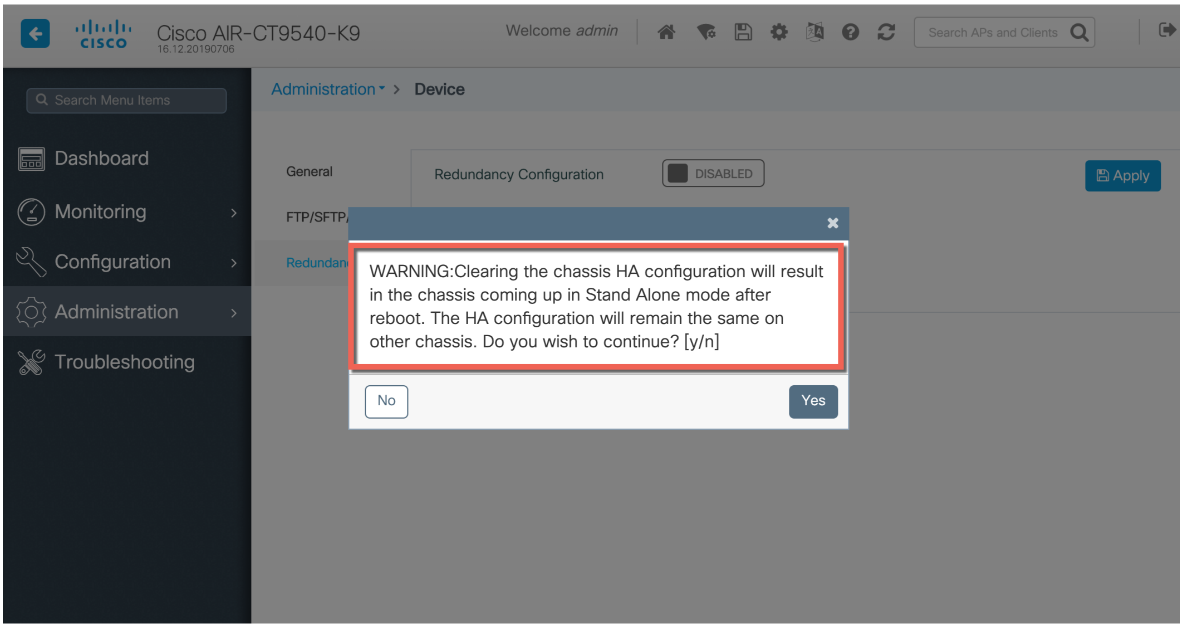
HA Unpairing Behavior

In release 16.10 and 16.11, when disjoining an HA pair by issuing the command 'clear chassis redundancy', the standby controller reboots and comes up with exactly the same configuration as the active controller, causing duplicate IP address error leading to the following messages:

```
WLC#sh log | i DUP
Mar 21 21:53:46.307 CET: %IP-4-DUPADDR: Duplicate address 120.0.0.1 on Vlan120, sourced by
d4c9.3ccc.f98b
Mar 21 21:54:16.947 CET: %IP-4-DUPADDR: Duplicate address 172.18.50.60 on
GigabitEthernet0, sourced by d4c9.3ccc.f981
```

The solution implemented in 16.12 and 17.1 is that after HA unpairing, the standby controller startup config and HA config will be cleared and standby will go to Day 0.

Before the command is executed, the user is prompted with the following warning on the active controller:



The same is seen on the CLI as well.

```
WLC#clear chassis redundancy
WARNING: Clearing the chassis HA configuration will result in both the chassis move into
Stand Alone mode. This involves reloading the standby chassis after clearing its HA
configuration and startup configuration which results in standby chassis coming up as a
totally clean after reboot. Do you wish to continue? [y/n]? [yes]:
*Apr  3 23:42:22.985: received clear chassis.. ha_supported:1yes
WLC#
*Apr  3 23:42:25.042: clearing peer startup config
*Apr  3 23:42:25.042: chkpt send: sent msg type 2 to peer..
*Apr  3 23:42:25.043: chkpt send: sent msg type 1 to peer..
*Apr  3 23:42:25.043: Clearing HA configurations
*Apr  3 23:42:26.183:  Successfully sent Set chassis mode msg for chassis 1.chasfs file
updated
*Apr  3 23:42:26.359: %IOSXE_REDUNDANCY-6-PEER_LOST: Active detected chassis 2 is no
longer standby
```

On the standby controller, the following messages indicate that the configuration is being cleared:

```
WLC-stby#
*Apr  3 23:40:40.537: mcprp_handle_spa_oir_tsm_event: subslot 0/0 event=2
*Apr  3 23:40:40.537:     spa_oir_tsm subslot 0/0 TSM: during state ready, got event
3(ready)
*Apr  3 23:40:40.537: @@@ spa_oir_tsm subslot 0/0 TSM: ready -> ready
*Apr  3 23:42:25.041: Removing the startup config file on standby
*Apr  3 23:42:26.466: Calling HA configs clear on standby
*Apr  3 23:42:26.466: Clearing HA configurations
*Apr  3 23:42:27.499:  Successfully sent Set chassis mode msg for chassis 2.chasfs file
updated
```

Note: To unpair the SSO pair when using RMI based config, use the “no” version of the RMI configuration followed command by reload:

```
WLC(config)# no redun-management interface <VLAN> chassis 1 address <RMI IP of chassis 1>
chassis 2 address <RMI IP of chassis 2>
```

LACP, PAGP support in SSO Pair

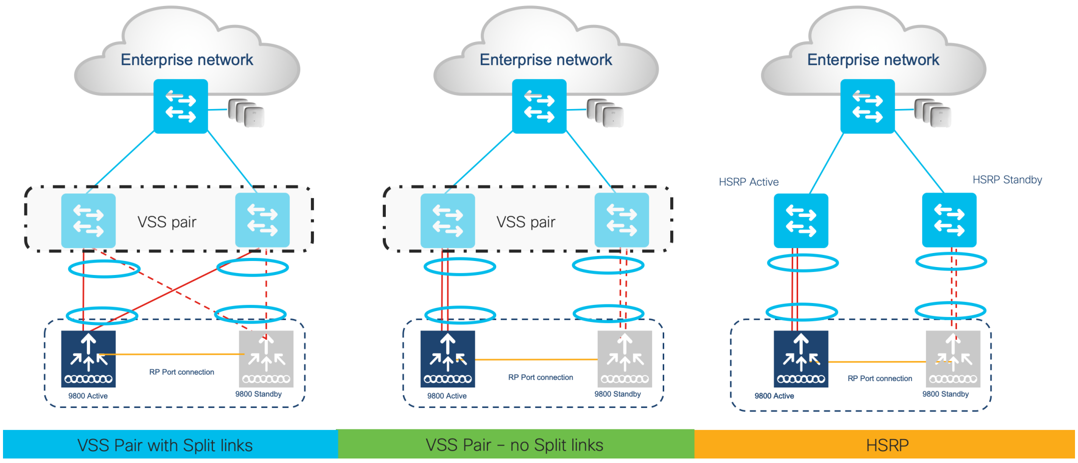
LACP protocol (IEEE 802.3ad) aggregates physical Ethernet interfaces by exchanging the Link Aggregation Control Protocol Data Units (LACPDUs) between two devices.

LACP, PAGP support is needed on SSO pair in order to have the ability to detect and monitor the link/connectivity failures on the standby controller and to have seamless transfer of client data traffic upon switchover (SSO). Prior to 17.1 only LAG mode ON was supported in SSO mode. With 17.1 both LACP (active and passive) and PAGP will be supported in SSO mode.

This feature is supported on Cisco Catalyst 9800-L, Cisco Catalyst 9800-40and Cisco Catalyst 9800-80 (including module ports).

Supported LACP, PAGP topologies

The following topologies are supported with SSO and LACP/PAGP



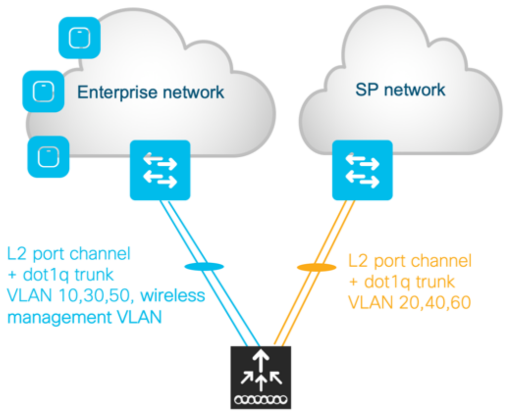
The following are not supported with LACP, PAGP topologies:

- Auto-LAG is not supported.
- C9800-CL and EWC on AP is not supported.
- L3 port-channel is not supported.

Multi-chassis Link Aggregation group

Starting with Release 17.2.1, Multi-chassis Link Aggregation Group is supported on a standalone as well as HA Pair of controllers. Multi-chassis LAG provides the capability to connect multiple uplinks from controller to separate uplink switches.

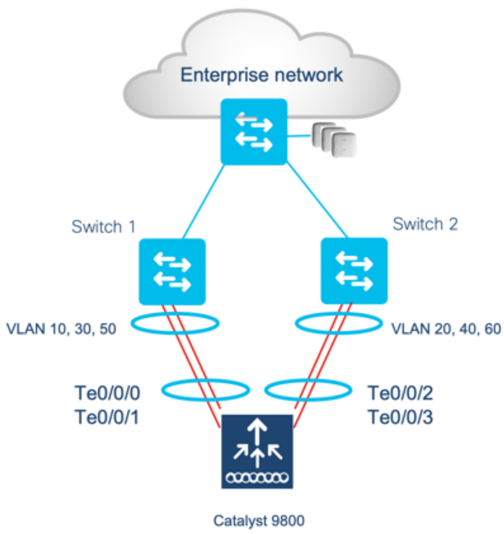
This enables flexibility in connecting controller(s) to switch infrastructure and VLAN-based traffic splitting when connected to a multi-switch topology, for e.g., to isolate Guest traffic on completely different switch/network from Enterprise traffic. Each LAG must be connected to a single switch and different VLANs must be assigned to different LAGs.

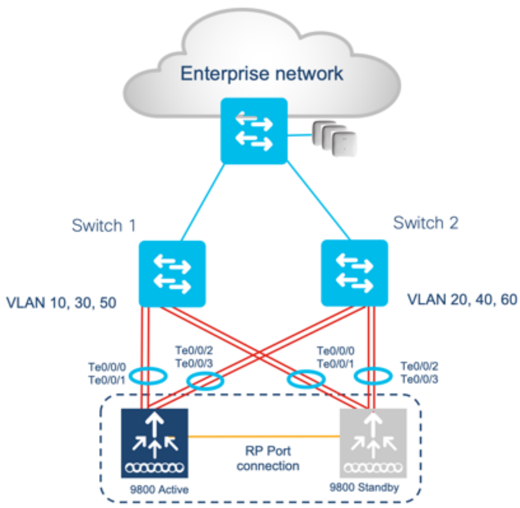


Note: It is the user’s configuration responsibility not to create a loop.

Supported Multi-chassis LAG topologies

- Multi-chassis LAG is supported with LAG mode ON and dynamic LAG (LACP and PAGP)
- Multi-chassis LAG is supported with a standalone controller as well as an HA pair as depicted below.





Note: Controller with multiple LAGs can be connected to a single switch, However, different VLANs must be connected to different LAGs

Supported Platforms:

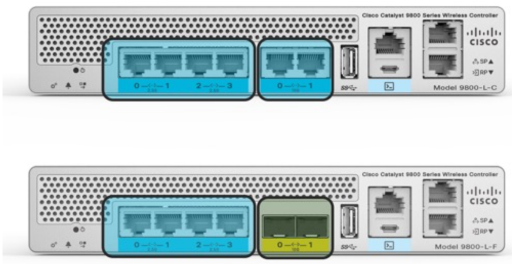
Multi-chassis LAG is supported on the following platforms:

- Catalyst 9800-L Wireless Controllers
- Catalyst 9800-40 Wireless Controllers
- Catalyst 9800-80 Wireless Controllers

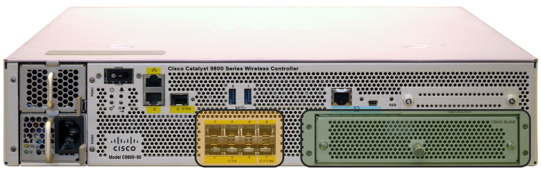
Supported LAG Port Grouping

Best practice is to have ports of same type and speed in the port channel

- 9800-L-C with 2.5G/1G and 10G/mGig ports in different port channels
- 9800-L-F with 2.5G/1G and 10G/1G Fiber ports in different port channels



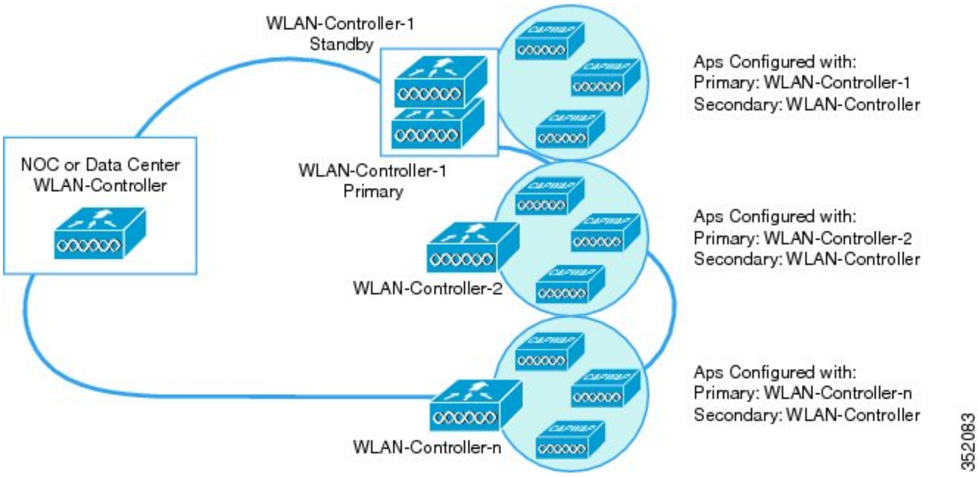
On the 9800-80 ports on Bay 0 and Bay 1 (modular slots) cannot be combined into the same port channel group. Best practice is to have ports of same slot in the port channel.



RMA process for a controller in an HA setup

- Disconnect and remove the standby controller from the HA pair in order to RMA the box.
- Prepare the new 9800 standby with the same configuration as the previous controller. This means the same software version, licensing level, wireless management interface, login credentials, and SSO configuration including RMI.
- Be sure to set the chassis priority to 1 so that it has a lower priority than the current active controller. Verify that a higher priority of 2 is set on the current active controller to make sure that the current active remains the active.
- Save the configuration and power off the new standby controller.
- Physically connect the cables on new standby which includes the uplink ports and redundancy port (RP).
- Power on the new standby controller. The new 9800 controller will pair up as standby with the current active controller.

N+1 with SSO Hybrid deployment



A hybrid topology of SSO redundant pair and N+1 primary, secondary and tertiary model is supported as shown above. The secondary controller at the DR site can be a Catalyst C9800-L, C9800-40 C9800-80 or C9800-CL Wireless controller. Access points failing back from Catalyst 9800 Wireless controller to CUWN controllers will re-download the code before joining the CUWN wireless controller and vice versa.

Standby Monitoring using RMI

This feature enables monitoring the health of the system on standby controller in an HA pair using programmatic interfaces (NETCONF/YANG, RESTCONF) and CLIs without going through the active controller. This includes monitoring parameters such as CPU, memory, interface status, PSU (Power Supply Unit) failure, fan failure and temperature. This feature is supported on the Cisco Catalyst 9800-CL Private cloud, 9800-L, 9800-40, and 9800-80 wireless controller.

Using the RMI interface, the user can:

- Connect to the IOS SSH server on port 22 to execute a select set of show CLIs.
- Connect to the NETCONF SSH server on port 830 and use programmatic interfaces to access NETCONF/YANG.
- Connect on the HTTPS port 443 and use programmatic interfaces using RESTCONF.

The user credentials can be configured locally for Local Authentication and External AAA server using RADIUS. SSH authentication shall be through user name and password. The standby controller does not run the PKI infrastructure to be able to handle certificate based authentication. External AAA servers shall be reachable through the default route which can be statically configured on the standby controller.

Syslog is supported on the standby controller as console logs.

Standby Monitoring using RMI IPv6 is supported starting release 17.3.2

Standby Monitoring CLIs

- To see power supply, fan and temperature status, the below CLI can be used on physical appliances. This output will be empty for virtual platforms.

```
Show environment
9800-stby#show environment summary

Number of Critical alarms:  0
Number of Major alarms:     0
Number of Minor alarms:     0

 Slot         Sensor         Current State   Reading        Threshold(Minor,Ma-
jor,Critical,Shutdown)
 ----------   -------------- --------------- ------------   -----------------------
----------------
 P0           Vin            Normal          218  V AC          na
 P0           Iin            Normal          1    A             na
 P0           Vout           Normal          12   V DC          na
 P0           Iout           Normal          20   A             na
 P0           Temp1          Normal          31   Celsius       (na ,na ,na ,na
)(Celsius)
 P0           Temp2          Normal          42   Celsius       (na ,na ,na ,na
)(Celsius)
 P0           Temp3          Normal          43   Celsius       (na ,na ,na ,na
)(Celsius)
 P1           Vin            Normal          0    V AC          na
 P1           Iin            Normal          0    A             na
 P1           Vout           Normal          0    V DC          na
 P1           Iout           Normal          1    A             na
 P1           Temp1          Normal          28   Celsius       (na ,na ,na ,na
)(Celsius)
 P1           Temp2          Normal          29   Celsius       (na ,na ,na ,na
)(Celsius)
 P1           Temp3          Normal          0    Celsius       (na ,na ,na ,na
)(Celsius)
 R0           VRRX1: VX1     Normal          751  mV            na
 R0           VRRX1: VX2     Normal          6937 mV            na
 R0           VRRX1: VX3     Normal          1217 mV            na
 R0           VRRX1: VX5     Normal          1222 mV            na
 R0           VRRX1: VP1     Normal          1705 mV            na
 R0           VRRX1: VP2     Normal          2489 mV            na
 R0           VRRX1: VP3     Normal          1300 mV            na
 R0           VRRX1: VP4     Normal          5070 mV            na
 R0           VRRX1: VH      Normal          11993mV            na
 R0           VRRX2: VX1     Normal          853  mV            na
 R0           VRRX2: VX4     Normal          1016 mV            na
 R0           VRRX2: VX5     Normal          1019 mV            na
 R0           VRRX2: VP1     Normal          3325 mV            na
 R0           VRRX2: VP3     Normal          1826 mV            na
 R0           VRRX2: VP4     Normal          1050 mV            na
 R0           VRRX2: VH      Normal          11987mV            na
 R0           VRRX3: VX1     Normal          994  mV            na
 R0           VRRX3: VX2     Normal          1002 mV            na
 R0           VRRX3: VX4     Normal          750  mV            na
 R0           VRRX3: VX5     Normal          751  mV            na
 R0           VRRX3: VP1     Normal          2477 mV            na
 R0           VRRX3: VP2     Normal          1197 mV            na
 R0           VRRX3: VP3     Normal          1517 mV            na
 R0           VRRX3: VP4     Normal          1514 mV            na
 R0           VRRX3: VH      Normal          11987mV            na
 R0           Temp: RCRX IN  Normal          26   Celsius       (52 ,57 ,62 ,73
)(Celsius)
 R0           Temp: RCRX OUT Normal          41   Celsius       (62 ,67 ,72 ,80
)(Celsius)
 R0           Temp: Yoda     Normal          47   Celsius       (71 ,76 ,81 ,90
)(Celsius)
 R0           Temp: XEPhy    Normal          49   Celsius
(110,120,130,140)(Celsius)
```

```
 R0          Temp: CPU Die    Normal          47    Celsius       (61 ,66 ,71 ,80
)(Celsius)
 R0          Temp: FC FANS    Fan Speed 40%   26    Celsius       (36 ,44 ,0  )(Cel-
sius)
```

- To get interface status on Standby controller, the below CLI can be used:

```
show ip interface brief
Eg.
9800-stby#show ip int brief
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet1       unassigned      YES unset  down                  down
GigabitEthernet0       unassigned      YES NVRAM  administratively down down
Capwap1                unassigned      YES unset  up                    up
Capwap2                unassigned      YES unset  up                    up
Capwap3                unassigned      YES unset  up                    up
Capwap4                unassigned      YES unset  up                    up
Capwap5                unassigned      YES unset  up                    up
Capwap6                unassigned      YES unset  up                    up
Capwap7                unassigned      YES unset  up                    up
Capwap8                unassigned      YES unset  up                    up
Capwap9                unassigned      YES unset  up                    up
Capwap10               unassigned      YES unset  up                    up
Vlan1                  unassigned      YES NVRAM  down                  down
Vlan56                 unassigned      YES unset  down                  down
Vlan111                111.1.1.85      YES NVRAM  up                    up
```

1. To see IOS task CPU on the standby, the CLI **show processes** can be used

```
9800-stby#show processes ?
  <1-2147483647>  IOS(d) Process Number
  cpu             Show CPU usage per IOS(d) process
  heapcheck       Show IOS(d) scheduler heapcheck configuration
  history         Show ordered IOS(d) process history
  memory          Show memory usage per IOS(d) process
  platform        Show information per IOS-XE process
  timercheck      Show IOS(d) processes configured for timercheck
  |               Output modifiers
  <cr>            <cr>
```

Standby Monitoring Programmatic Interfaces

The CPU, memory and interface status on standby controller can be monitored programmatic interfaces. Here is the list of operational models required for this purpose:

- **Cisco-IOS-XE-device-hardware-oper.yang**: This has serial number for all FRUs in the device, including chassis. It also has information about all hardware in the system.
- **Cisco-IOS-XE-process-cpu-oper.yang**: This has CPU utilization averages over intervals of past 1 min, 5 min, 5 seconds, and also per process CPU stats for IOS tasks.
- **Cisco-IOS-XE-platform-software-oper.yang**: This gives Average CPU utilization of 5-second interval and allocated memory for the processes.

2. **Cisco-IOS-XE-process-memory-oper.yang**: This gives per process memory utilization.

- **Cisco-IOS-XE-interfaces-oper.yang**: This has interface operational data including state and stats. It has a lot of other operational data about interfaces also.

Steps to monitor the standby controller using SSH to RMI IPv4

1. Enable SSH on the active controller. In order to do that, it is required to generate rsa key

```
9800(config)#crypto key generate rsa
% You already have RSA keys defined named ak_vewlc_small.cisco.com.
% Do you really want to replace them? [yes/no]: yes
Choose the size of the key modulus in the range of 2048 to 4096 for your
  General Purpose Keys. Choosing a key modulus greater than 512 may take
  a few minutes.

How many bits in the modulus [2048]: 2048
% Generating 2048 bit RSA keys, keys will be non-exportable...
[OK] (elapsed time was 0 seconds)
9800(config)#
```

Configure Local AAA or External AAA (RADIUS) with local AAA fallback as shown below.

```
line vty 0 4
 password Cisco
 authorization exec DEVICE_ADMIN
 login authentication DEVICE_ADMIN
 length 0
 transport input ssh

line vty 5 15
 password Cisco
 authorization exec DEVICE_ADMIN
 login authentication DEVICE_ADMIN
 transport input telnet ssh
 transport output telnet ssh

aaa authentication login DEVICE_ADMIN group AAA_GROUP_ISE1 local
aaa authorization exec DEVICE_ADMIN group AAA_GROUP_ISE1 local
aaa group server radius AAA_GROUP_ISE1
 server name ISE1
radius server ISE1
 address ipv4 <RMI IP> auth-port 1812 acct-port 1813
 key <key>
```

Note: TACACS is not supported for standby. Make sure “LOCAL” is added in the method list. So user will be authenticated locally for standby.

```
aaa authentication login VTY_authen_tacacs group tacacs_ise_group local
aaa authentication login VTY_authen_tacacs group tacacs_ise_group local
```

2. Make sure default route is configured for management VLAN.

```
ip route <Destination prefix> <Destination prefix mask> <Forwarding router's ad-
dress>
```

3. Login to the standby controller using the standby controller’s RMI IP address

```
ssh <username>@<RMI IP>
Password:
```

Note: To use Netconf-YANG SSH use the command:
ssh *<username>@<RMI IP>* -p 830

Only the default port of 830 can be used for Netconf-YANG SSH

4. Execute the commands **show environment summary, show processes, show ip interface brief** to view the CPU, memory, interface status, PSU (Power Supply Unit) failure, fan failure and temperature.

Command for Standby Monitoring using RESTCONF

GET request:

```
curl --request GET --url https://<Standby RMI IP>:443/restconf/data/Cisco-IOS-XE-native:native/hostname --header 'accept: application/yang-data+json' --header 'cache-control: no-cache' --header 'content-type: application/yang-data+json' -k -u username:password
```

eg.

```
$curl --request GET --url https://<Standby RMI IP>:443/restconf/data/Cisco-IOS-XE-native:native/hostname --header 'accept: application/yang-data+json' --header 'cache-control: no-cache' --header 'content-type: application/yang-data+json' -k -u username:password
{
"Cisco-IOS-XE-native:hostname": "Catalyst 9800 HA2"
}
```

PUT request is not supported for the standby and will return an access-denied error.

Caveats of Standby Monitoring

- SNMP support on the standby controller is not supported
- External syslog server on the standby controller is not supported
- SSH to IOS will generate syslogs on standby console. NetConf SSH login will generate syslogs on the active console.
- Standby monitoring using the service port is not supported
- Accounting on standby controller is not supported
- External AAA with TACACS is not supported
- Rad-Sec is not supported
- Embedded controller on Switch does not support this feature
- Cannot do standby monitoring on controller in Active-Recovery mode since all its interfaces will be in Admin Down state.