



High Availability

- [High Availability, on page 1](#)
- [Redundancy management interfaces, on page 14](#)

High Availability

A high availability feature is a wireless controller capability that

- reduces network downtime by enabling seamless failover between active and standby controllers,
- preserves AP and client connectivity through stateful switchover by maintaining CAPWAP tunnels and client sessions during failover, and
- mirrors AP and client databases from the active controller to the standby controller to prevent APs from entering discovery state and avoid client disconnections.

Feature history for High Availability

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

Table 1: Feature history

Release	Feature	Feature information
Cisco IOS XE 17.18.1	Enhanced Gateway Reachability Statistics	<p>Improves visibility into gateway reachability and provides detailed statistics for ICMP, ARP, and ND probes. This feature enables simplified troubleshooting, greater transparency, and more reliable diagnostics for HA and RMI functionality.</p> <p>This command is introduced:</p> <ul style="list-style-type: none"> • show platform software rif-mgr chassis { active standby} r0 gateway-statistics <p>This command is modified:</p> <ul style="list-style-type: none"> • show platform software rif-mgr chassis { active standby} r0 resource-status
Cisco IOS XE 17.1.1s	Redundant Management Interface	The Redundancy Management Interface (RMI) is used as a secondary link between the active and standby controllers. This interface is the same as the Wireless Management Interface and the IP address on this interface is configured in the same subnet as the Wireless Management Interface.

Additional reference information

High availability enables seamless controller failover, ensuring that APs and clients remain connected during controller outages. These notes and recommendations apply to HA deployments:

-
- When the controller operates as a spanning tree host, configure portfast trunk on the uplink switch to ensure faster convergence. Use **spanning-tree port type edge trunk** or **spanning-tree portfast trunk**.
- You can configure FIPS in an HA setup. For information, see the [Configuring FIPS in HA Setup](#).
- Do not configure the secondary IPv4 address. The controller uses the IPv4 secondary address internally for RMI purposes.

Configure only one management IPv6 address on the Wireless Management Interface (WMI). The controller uses any secondary address for RMI-IPv6.

Configuring more than one management IPv4 address or more than one management IPv6 address on the WMI may cause unpredictable behavior.

During a failover event, only one CAPWAP tunnel is maintained between APs and the active controller. This ensures zero downtime for client services and no SSID outages. Database mirroring prevents APs from entering the discovery state and ensures that clients remain connected without interruption.

Prerequisites for High Availability

To ensure high availability, configure interfaces properly, select the appropriate HA port, and meet latency, bandwidth, and MTU requirements for RP links.

External interfaces and IPs

All interfaces are configured on the Active box and synchronized with the Standby box. Therefore, the same set of interfaces is present on both controller s.

External nodes connect to the same IP addresses regardless of which controller they are connected to.

APs, clients, DHCP servers, Cisco Prime Infrastructure, Cisco Catalyst Center, and Cisco Identity Services Engine (ISE) servers, as well as other controller members in the mobility group, always connect to the same IP address.

The SSO switchover is transparent to these devices. However, if TCP connections exist from external nodes to the controller , reset and reestablish those connections.

HA interfaces

The HA interface provides:

- Provides connectivity between the controller pair before an IOSd comes up,
- provides IPC transport across the controller pair, and
- enables redundancy across control messages exchanged between the controller pair. The control messages include HA role resolution, keepalive messages, notifications, HA statistics, and similar messages.

You can select an SFP or RJ-45 connection for the HA port. Supported Cisco SFPs are:

- GLC-SX-MMD
- GLC-LH-SMD

HA operates when either an SFP or RJ-45 connection is present between the two controllers. SFP connectivity takes priority over RJ-45 connectivity.

If you connect an SFP link while RJ-45 HA is active, the HA pair restarts. The restart occurs even if the SFP link is not connected.

HA operates when either an SFP or RJ-45 connection is present between the two controllers. If you connect an SFP link while RJ-45 HA is active, the HA pair restarts. The restart occurs even if the SFP link is not connected.



Note Connect RP links using switches to enable controller HA. Keep the round-trip time between the two controllers under 80 milliseconds.

High Availability restrictions

- Wait until configuration synchronization completes on the standby controller. Before initiating a fail-safe Stateful Switchover (SSO), ensure that the standby controller has been powered on for sufficient time

(up to 24 minutes [up to 1,440 seconds] on some platforms) to achieve readiness. Use the **show wireless stats redundancy config database** command to view database statistics.

- During a switchover in local mode, NBAR engine flow states are lost. As a result, classification restarts and may lead to incorrect packet classification.
- You can use HA connections only with IPv4.
- When you perform a switchover or an active reload, the high-availability link goes down on the new primary controller.
-
- You cannot access the web UI from the standby RMI interface.
-
-
- Configure two HA interfaces, RMI and RP, on the same subnet. Do not share this subnet with any other interfaces on the device.
- After a switchover, you must re-establish any TCP session because synchronization is not possible.
- Client SSO does not address clients that have not reached the RUN state. These clients are removed after a switchover.
- Statistics tables are not synchronized from the active controller to the standby controller.
- Creating a machine snapshot of a VM hosting controller HA interfaces is not supported. This action may lead to a crash in the HA controller.
- Clients that are not in RUN state are reauthenticated after a switchover.
- Application classification may not be retained after SSO:
 - AVC limitation—After a switchover, context transfer or synchronization to the standby controller does not occur. The new active flow must be relearned. AVC QoS does not take effect during classification failure.
 - A voice call cannot be recognized after a switchover because a voice policy is based on RTP or RTCP protocol.
 - Auto QoS does not work due to AVC limitation.
- For virtual platforms, pair the active controller and the standby controller with the same interface. For hardware appliances, use a dedicated HA port.
- You can synchronize static IP addressing to the standby controller, but you cannot use the IP address from the standby controller.
- You can map a dedicated HA port to a 1-gigabit (1,000 Mbps) interface only.
- To use EtherChannels in HA mode in releases up to Cisco IOS XE Gibraltar 16.12.x, ensure that the channel mode is set to On.
- EtherChannel Auto-mode is not available in HA mode in releases up to Cisco IOS XE Gibraltar 16.12.x.
- LACP and PAGP protocols cannot be used in HA mode in releases up to Cisco IOS XE Gibraltar 16.12.x.

- When the controller operates as a host for spanning tree, configure portfast trunk on the uplink switch using **spanning-tree port type edge trunk** or **spanning-tree portfast trunk** command to ensure faster convergence.
- The **clear chassis redundancy** and **write erase** commands do not reset the chassis priority to the default value.
- While configuring devices in HA, ensure that members do not have wireless trustpoints with the same name but different keys. If you form an HA pair between two standalone controllers with mismatched trustpoints, the wireless trustpoint does not come up after SSO. The *rsa keypair* file exists but is incorrect because the *nvrाम:private-config* file is not synchronized with the actual *WLC_WLC_TP* key pair.
- Before forming HA, delete existing certificates and keys from each controller that was previously deployed as standalone. This is a best practice.
- Do not configure the WLAN or WLAN policy after a switchover while recovery is in progress. Doing so may cause the controller to crash.
- After a switchover, clients that are not in RUN state and not connected to an AP are removed after 300 seconds (5 minutes).

Best practices for RP port configuration

When you configure RP ports, use these best practices:

- Ensure that the Local and Remote IP addresses are in the same subnet.
- Use the 169.254.X.X/16 subnet, deriving the last two octets from the management interface.
- Do not use the 10.10.10.x/24 subnet for the RP port.
- For more information about RMI+RP chosen as the redundancy method, see [Information About Redundancy Management Interface](#) .

Configure High Availability (CLI)

Set up high availability for network redundancy and automatic failover between devices using the CLI.

Before you begin

Ensure that the active and standby controllers use the same mode—either Install mode or Bundle mode—and the same image version. Use Install mode.

Procedure

-
- Step 1** (Optional) Configure the priority of the device.

Example:

```
Device# chassis chassis-num priority chassis-priority
```

Note

From Cisco IOS XE 16.12.x onward, a device reload is not required for the chassis priority to take effect.

- *chassis-num* —Enter the chassis number (range: one to two).
- *chassis-priority* —Enter the chassis priority (range: one to two; default: one).

Note

If both devices boot up simultaneously, the device with the higher priority (2) becomes active. The other device becomes standby. If both devices have the same priority, the device with the smaller MAC address becomes active, and the peer device becomes standby.

Step 2 Set the chassis high-availability parameters.

Example:

```
Device# chassis redundancy ha-interface GigabitEthernet num local-ip local-chassis-ip-addr
network-mask remote-ip remote-chassis-ip-addr
```

Example:

```
Device# chassis redundancy ha-interface
GigabitEthernet 2 local-ip 4.4.4.1 /24 remote-ip 4.4.4.2
```

- *num* —GigabitEthernet interface number (range: zero to 32).
- *local-chassis-ip-addr* —Enter the IP address of the local chassis high-availability interface.
- *network-mask* —Enter the network mask or prefix length in slash nn or dotted decimal format.
- *remote-chassis-ip-addr* —Enter the remote chassis IP address.

Step 3 Configure the peer keepalive timeout value.

Example:

```
Device# chassis redundancy keep-alive timer timer
```

Set the time interval in multiples of 100 milliseconds (ms). Enter one for the default value.

Step 4 Set the peer keepalive retry value that determines when the system considers the peer down.

Example:

```
Device# chassis redundancy keep-alive retries retry-value
```

The default value is five.

After you complete these steps, high availability is configured between two devices. The system uses device priorities, high-availability interfaces, and keepalive parameters to ensure redundancy and seamless failover if a device fails.

Disable High Availability

When you disable high availability, all HA-related parameters are removed and the controller returns to stand-alone mode.

- Use **clear chassis redundancy** with the RP method to clear the local IP, remote IP, HA interface, mask, timeout, and priority.
- Use **no redun-management interface vlan chassis** with the RMI method.

- After you unpair the controllers, the startup and HA configuration of the standby controller are cleared, and it enters Day zero state.

If you configure the controller using the RP method for SSO, use this command to clear all HA-related parameters: local IP, remote IP, HA interface, mask, timeout, and priority:

- **clear chassis redundancy**

If you configure the controller using the RMI method, use this command:

- **no redun-management interface vlan chassis**



Note This command is not supported on these models:

- Cisco Catalyst CW9800H1 Wireless Controller .
- Cisco Catalyst CW9800H2 Wireless Controller .
- Cisco Catalyst CW9800M Wireless Controller .



Note Reload your devices to apply the changes.

Before you execute the command, you see this warning on the active controller:

```
Device# 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:lyes
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, these messages indicate that the configuration is being cleared:

```
Device-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
!Standby controller is reloaded after clearing the chassis.
```

Verify high availability configurations

To view the HA configuration details, use this command:

```

Device# show romvar
ROMMON variables:
LICENSE_BOOT_LEVEL =
MCP_STARTUP_TRACEFLAGS = 00000000:00000000
BOOTLDR =
CRASHINFO = bootflash:crashinfo_RP_00_00_20180202-034353-UTC
STACK_1_1 = 0_0
CONFIG_FILE =
BOOT =
bootflash:boot_image_test,1;bootflash:boot_image_good,1;bootflash:rp_super_universalk9.vwlc.bin,1;

RET_2_RTS =
SWITCH_NUMBER = 1
CHASSIS_HA_REMOTE_IP = 10.0.1.9
CHASSIS_HA_LOCAL_IP = 10.0.1.10
CHASSIS_HA_LOCAL_MASK = 255.255.255.0
CHASSIS_HA_IFNAME = GigabitEthernet2
CHASSIS_HA_IFMAC = 00:0C:29:C9:12:0B
RET_2_RCALTS =
BSI = 0
RANDOM_NUM = 647419395

```

Verify AP or client SSO statistics

To view the AP SSO statistics, use this command:

```

Device# show wireless stat redundancy statistics ap-recovery wnc all
AP SSO Statistics

```

Inst	Timestamp	Dura (ms)	#APs	#Succ	#Fail	Avg (ms)	Min (ms)	Max (ms)
0	00:06:29.042	98	34	34	0	2	1	35
1	00:06:29.057	56	33	30	3	1	1	15
2	00:06:29.070	82	33	33	0	2	1	13

Statistics:

```

WNCD Instance : 0
No. of AP radio recovery failures : 0
No. of AP BSSID recovery failures : 0
No. of CAPWAP recovery failures : 0
No. of DTLS recovery failures : 0
No. of reconcile message send failed : 0
No. of reconcile message successfully sent : 34
No. of Mesh BSSID recovery failures: 0
No. of Partial delete cleanup done : 0
.
.
.

```

To view the Client SSO statistics, use this command:

```

Device# show wireless stat redundancy client-recovery wncd all
Client SSO statistics
-----

```

```

WNCD instance : 1
Reconcile messages received from AP : 1
Reconcile clients received from AP : 1
Recreate attempted post switchover : 1
Recreate attempted by SANET Lib : 0
Recreate attempted by DOT1x Lib : 0

```

```

Recreate attempted by SISF Lib : 0
Recreate attempted by SVC CO Lib : 1
Recreate attempted by Unknown Lib : 0
Recreate succeeded post switchover : 1
Recreate Failed post switchover : 0
Stale client entries purged post switchover : 0

Partial delete during heap recreate : 0
Partial delete during force purge : 0
Partial delete post restart : 0
Partial delete due to AP recovery failure : 0
Partial delete during reconciliation : 0

Client entries in shadow list during SSO : 0
Client entries in shadow default state during SSO : 0
Client entries in poison list during SSO : 0

Invalid bssid during heap recreate : 0
Invalid bssid during force purge : 0
BSSID mismatch with shadow rec during reconciliation : 0
BSSID mismatch with shadow rec reconciliation(WGB client): 0
BSSID mismatch with dot11 rec during heap recreate : 0

AID mismatch with dot11 rec during force purge : 0
AP slotid mismatch during reconciliation : 0
Zero aid during heap recreate : 0
AID mismatch with shadow rec during reconciliation : 0
AP slotid mismatch shadow rec during reconciliation : 0
Client shadow record not present : 0

```

To view the mobility details, use this command:

```

Device# show wireless stat redundancy client-recovery mobilityd
Mobility Client Deletion Reason Statistics
-----
Mobility Incomplete State : 0
Inconsistency in WNCd & Mobility : 0
Partial Delete : 0

General statistics
-----
Cleanup sent to WNCd, Missing Delete case : 0

```

To view the Client SSO statistics for SISF, use this command:

```

Device# show wireless stat redundancy client-recovery sisf
Client SSO statistics for SISF
-----
Number of recreate attempted post switchover : 1
Number of recreate succeeded post switchover : 1
Number of recreate failed because of no mac : 0
Number of recreate failed because of no ip : 0
Number of ipv4 entry recreate success : 1
Number of ipv4 entry recreate failed : 0
Number of ipv6 entry recreate success : 0
Number of ipv6 entry recreate failed : 0
Number of partial delete received : 0
Number of client purge attempted : 0
Number of heap and db entry purge success : 0
Number of purge success for db entry only : 0
Number of client purge failed : 0
Number of garp sent : 1
Number of garp failed : 0
Number of IP entries validated in cleanup : 0

```

```

Number of IP entry address errors in cleanup      : 0
Number of IP entry deleted in cleanup            : 0
Number of IP entry delete failed in cleanup      : 0
Number of IP table create callbacks on standby  : 0
Number of IP table modify callbacks on standby  : 0
Number of IP table delete callbacks on standby  : 0
Number of MAC table create callbacks on standby : 1
Number of MAC table modify callbacks on standby : 0
Number of MAC table delete callbacks on standby : 0

```

To view the HA redundancy summary, use this command:

```

Device# show wireless stat redundancy summary
HA redundancy summary
-----

AP recovery duration (ms)      : 264
SSO HA sync timer expired     : No

```

Verify high availability

Table 2: Commands for monitoring chassis and redundancy

Command Name	Description
show chassis	<p>Displays the chassis information.</p> <p>Note When the peer timeout and retries are configured, the show chassis ha-status command output may show incorrect values.</p> <p>To check the peer keep-alive timer and retries, use these commands:</p> <ul style="list-style-type: none"> • show platform software stack-mgr chassis active r0 peer-timeout • show platform software stack-mgr chassis standby r0 peer-timeout
show redundancy	Displays details about Active box and Standby box.
show redundancy switchover history	Displays the switchover counts, switchover reason, and the switchover time.

To start the packet capture in the redundancy HA port (RP), use these commands:

- test wireless redundancy packet dump start
- test wireless redundancy packet dump stop
- test wireless redundancy packet dump start filter port 2300

```

Device# test wireless redundancy packetdump start
Redundancy Port PacketDump Start
Packet capture started on RP port.

```

```

Device# test wireless redundancy packetdump stop
Redundancy Port PacketDump Start
Packet capture started on RP port.

```

```

Redundancy Port PacketDump Stop
Packet capture stopped on RP port.
Device# dir bootflash:
Directory of bootflash:/
1062881 drwx      151552  Oct 20 2020 23:15:25 +00:00  tracelogs
47      -rw-       20480  Oct 20 2020 23:15:24 +00:00  haIntCaptureLo.pcap
1177345 drwx      4096   Oct 20 2020 19:56:14 +00:00  certs
294337  drwx      8192   Oct 20 2020 19:56:05 +00:00  license_evlog
15      -rw-       676    Oct 20 2020 19:56:01 +00:00  vlan.dat
14      -rw-       30     Oct 20 2020 19:55:16 +00:00  throughput_monitor_params
13      -rw-     134808  Oct 20 2020 19:54:57 +00:00  memleak.tcl
1586145 drwx      4096   Oct 20 2020 19:54:45 +00:00  .inv
1103761 drwx      4096   Oct 20 2020 19:54:39 +00:00  dc_profile_dir
17      -r--      114    Oct 20 2020 19:54:17 +00:00  debug.conf
1389921 drwx      4096   Oct 20 2020 19:54:17 +00:00  .installer
46      -rw-     1104760207 Oct 20 2020 19:26:41 +00:00  leela_katar_rping_test.SSA.bin
49057   drwx      4096   Oct 20 2020 16:11:21 +00:00  .prst_sync
45      -rw-     1104803200 Oct 20 2020 15:39:19 +00:00
C9800-I-universalk9_wlc.2020-10-20_14.57_yavadhan.SSA.bin
269809  drwx      4096   Oct 19 2020 23:41:49 +00:00  core
44      -rw-     1104751981 Oct 19 2020 17:42:12 +00:00
C9800-I-universalk9_wlc.BLD_POLARIS_DEV_LATEST_20201018_053825_2.SSA.bin
43      -rw-     1104286975  Oct 16 2020 12:05:47 +00:00
C9800-I-universalk9_wlc.BLD_POLARIS_DEV_LATEST_20201010_001654_2.SSA.bin

```

```

Device# test wireless redundancy packetdump start filter port 2300
Redundancy Port PacketDump Start
Packet capture started on RP port with port filter 2300.

```

To check connection between the two HA Ports (RP) and check if there are any drops, delays, or jitter in the connection, use this command:

```

Device# test wireless redundancy rping
Redundancy Port ping
PING 169.254.64.60 (169.254.64.60) 56(84) bytes of data.
64 bytes from 169.254.64.60: icmp_seq=1 ttl=64 time=0.083 ms
64 bytes from 169.254.64.60: icmp_seq=2 ttl=64 time=0.091 ms
64 bytes from 169.254.64.60: icmp_seq=3 ttl=64 time=0.074 ms

--- 169.254.64.60 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2041ms
rtt min/avg/max/mdev = 0.074/0.082/0.091/0.007 ms
test wireless redundancy

```

To see the HA port interface setting status, use the **show platform hardware slot R0 ha_port interface stats** command.

```

Device# show platform hardware slot R0 ha_port interface stats
HA Port
ha_port  Link encap:Ethernet  HWaddr 70:18:a7:c8:80:70
         UP BROADCAST MULTICAST  MTU:1500  Metric:1
         RX packets:0 errors:0 dropped:0 overruns:0 frame:0
         TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1000
         RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
         Memory:e0900000-e0920000

Settings for ha_port:
Supported ports:          [ TP ]
Supported link modes:    10baseT/Half 10baseT/Full
                        100baseT/Half 100baseT/Full
                        1000baseT/Full
Supported pause frame use: Symmetric
Supports auto-negotiation: Yes

```

```

Supported FEC modes:      Not reported
Advertised link modes:   10baseT/Half 10baseT/Full
                          100baseT/Half 100baseT/Full
                          1000baseT/Full
Advertised pause frame use: Symmetric
Advertised auto-negotiation: Yes
Advertised FEC modes:    Not reported
Speed:                   Unknown!
Duplex:                   Unknown! (255)
Port:                     Twisted Pair
PHYAD:                    1
Transceiver:              internal
Auto-negotiation:        on
MDI-X:                    off (auto)
Supports Wake-on:        pumbg
Wake-on:                  g
Current message level:    0x00000007 (7)
                          drv probe link
Link detected:           no

```

```

NIC statistics:
rx_packets:              0
tx_packets:              0
rx_bytes:                 0
tx_bytes:                 0
rx_broadcast:            0
tx_broadcast:            0
rx_multicast:            0
tx_multicast:            0
multicast:               0
collisions:              0
rx_crc_errors:           0
rx_no_buffer_count:      0
rx_missed_errors:        0
tx_aborted_errors:       0
tx_carrier_errors:       0
tx_window_errors:        0
tx_abort_late_coll:      0
tx_deferred_ok:          0
tx_single_coll_ok:       0
tx_multi_coll_ok:        0
tx_timeout_count:        0
rx_long_length_errors:   0
rx_short_length_errors:  0
rx_align_errors:         0
tx_tcp_seg_good:         0
tx_tcp_seg_failed:       0
rx_flow_control_xon:     0
rx_flow_control_xoff:    0
tx_flow_control_xon:     0
tx_flow_control_xoff:    0
rx_long_byte_count:      0
tx_dma_out_of_sync:      0
tx_smbus:                 0
rx_smbus:                 0
dropped_smbus:           0
os2bmc_rx_by_bmc:        0
os2bmc_tx_by_bmc:        0
os2bmc_tx_by_host:       0
os2bmc_rx_by_host:       0
tx_hwtstamp_timeouts:    0
rx_hwtstamp_cleared:     0
rx_errors:                0
tx_errors:                0

```

```
tx_dropped: 0
rx_length_errors: 0
rx_over_errors: 0
rx_frame_errors: 0
rx_fifo_errors: 0
tx_fifo_errors: 0
tx_heartbeat_errors: 0
tx_queue_0_packets: 0
tx_queue_0_bytes: 0
tx_queue_0_restart: 0
tx_queue_1_packets: 0
tx_queue_1_bytes: 0
tx_queue_1_restart: 0
rx_queue_0_packets: 0
rx_queue_0_bytes: 0
rx_queue_0_drops: 0
rx_queue_0_csum_err: 0
rx_queue_0_alloc_failed:0
rx_queue_1_packets: 0
rx_queue_1_bytes: 0
rx_queue_1_drops: 0
rx_queue_1_csum_err: 0
rx_queue_1_alloc_failed:0
```

Configure a switchover

Enable manual failover to the standby unit and verify that high availability and redundancy work as expected.

Procedure

Force a failover to the standby unit by entering this command:

Example:

```
Device#redundancy force-switchover
```

After you enter this command, the standby controller takes the active role. The active controller reloads and becomes the standby controller. Use this command to test high availability cluster stability and confirm that switchovers work as expected.

Note

Use only the recommended command to test switchovers between Cisco Catalyst 9800 series wireless controllers. Using other commands, such as **reload slot X** (where X is the active controller), might cause unexpected behavior.

Note

In a scaled environment, avoid performing an immediate switchover after modifying WLAN or policy profile configurations. Doing so might cause unexpected behavior.

The system completes the switchover. The standby controller becomes active, and the previous active controller reloads to become standby. The switchover verifies high availability cluster stability.

Redundancy management interfaces

A redundancy management interface is a high availability feature that

- enables pairing scenarios and gateway monitoring for Cisco Catalyst 9800 Series Wireless Controllers
- supports both IPv4 and IPv6 dual stack environments, and
- facilitates dynamic pairing, upgrade/downgrade behaviors, ARP handling, and AAA integration.

Gateway-monitoring configuration verification

Verify the status of the gateway-monitoring configuration on active and standby controllers using specific show commands.

To verify the status of the gateway-monitoring configuration on an active controller, run this command:

```
Device# show redundancy states

my state = 13 -ACTIVE
peer state = 8 -STANDBY HOT
Mode = Duplex
Unit = Primary
Unit ID = 1

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

client count = 129
client_notification_TMR = 30000 milliseconds
RF debug mask = 0x0
Gateway Monitoring = Disabled
Gateway monitoring interval = 8 secs
```

To verify the status of the gateway-monitoring configuration on a standby controller, run this command:

```
Device-stby# show redundancy states

my state = 8 -STANDBY HOT
peer state = 13 -ACTIVE
Mode = Duplex
Unit = Primary
Unit ID = 2

Redundancy Mode (Operational) = sso
Redundancy Mode (Configured) = sso
Redundancy State = sso
Maintenance Mode = Disabled
Manual Swact = cannot be initiated from this the standby unit
Communications = Up

client count = 129
client_notification_TMR = 30000 milliseconds
RF debug mask = 0x0
Gateway Monitoring = Disabled
Gateway monitoring interval = 8 secs
```

RMI IPv4 configuration verification

Verify the RMI IPv4 configuration.

```
Device# show running-config interface vlan management-vlan

Building configuration...

Current configuration : 109 bytes
!
interface Vlan90
ip address 9.10.90.147 255.255.255.0 secondary
ip address 9.10.90.41 255.255.255.0
end
```

To verify the interface configuration for a standby controller, use this command:

```
Device-stby# show running-config interface vlan 90

Building configuration...

Current configuration : 62 bytes
!
interface Vlan90
ip address 9.10.90.149 255.255.255.0
end
```

To verify the chassis redundancy management interface configuration for an active controller, use this command:

```
Device# show chassis rmi

Chassis/Stack Mac Address : 000c.2964.1eb6 - Local Mac Address
Mac persistency wait time: Indefinite
      H/W Current
Chassis# Role      Mac Address      Priority  Version  State  IP              RMI-IP
-----
*1      Active   000c.2964.1eb6   1        V02     Ready  169.254.90.147  9.10.90.147
2       Standby  000c.2975.3aa6   1        V02     Ready  169.254.90.149  9.10.90.149
```

To verify the chassis redundancy management interface configuration for a standby controller, use this command:

```
Device-stby# show chassis rmi

Chassis/Stack Mac Address : 000c.2964.1eb6 - Local Mac Address
Mac persistency wait time: Indefinite
      H/W Current
Chassis# Role      Mac Address      Priority  Version  State  IP              RMI-IP
-----
1       Active   000c.2964.1eb6   1        V02     Ready  169.254.90.147  9.10.90.147
*2     Standby  000c.2975.3aa6   1        V02     Ready  169.254.90.149  9.10.90.149
```

To verify the ROMMON variables on an active controller, use this command:

```
Device# show romvar | include RMI

RMI_INTERFACE_NAME = Vlan90
RMI_CHASSIS_LOCAL_IP = 9.10.90.147
RMI_CHASSIS_REMOTE_IP = 9.10.90.149
```

To verify the ROMMON variables on a standby controller, use this command:

```
Device-stby# show romvar | include RMI

RMI_INTERFACE_NAME = Vlan90
RMI_CHASSIS_LOCAL_IP = 9.10.90.149
RMI_CHASSIS_REMOTE_IP = 9.10.90.147
```

To verify the switchover reason, use this command:

```
Device# show redundancy switchover history
```

Index	Previous active	Current active	Switchover reason	Switchover time
1	2	1	Active lost GW	17:02:29 UTC Mon Feb 3 2020

Redundancy management interfaces

A redundancy management interface is a network interface that

- acts as a secondary link between active and standby wireless controllers,
- enables resource health information exchange (such as gateway reachability), and
- assists in the detection of dual-active controller conditions to maintain high availability.

The RMI might trigger a switchover based on the gateway status on the controllers.

Subdefinitions:

- **Active controller:** Uses the management IP as the primary address and RMI as the secondary IPv4 address on the management VLAN. RMI configuration is automatic.
Analogy: Like a city's current mayor who's in charge, but always ready to hand over leadership if needed.
- **Standby controller:** Has the RMI IP as the primary address; upon switchover, roles and addresses are swapped.
Analogy: Like a vice-mayor who takes the mayor's seat (with all responsibilities and keys) when the mayor is away.
- **RP (Redundancy Port):** The main dedicated physical link for state and configuration synchronization between active and standby controllers; loss of both RP and RMI links results in high availability (HA) failures.
(Analogy: Like the main road connecting two city offices, critical for daily business and coordination).
- **WMI (Wireless Management Interface):** The main management interface for controller operations and communications; shares its subnet with the RMI and may serve as a source address for certain types of traffic such as AAA packets.
Analogy: Like a city's public headquarters, used for both regular operations and official correspondence.
- **RMI (Redundancy Management Interface):** A dedicated network interface that serves as a secondary communication path between controllers for exchanging resource health information, detecting dual-active conditions, and monitoring gateway reachability; shares the same subnet as the Wireless Management Interface (WMI).

Analogy: Like an emergency side road connecting two city halls, used for urgent official communication.

- **HA (High Availability):** A deployment setup where controllers operate as an active-standby pair to ensure uninterrupted wireless services; relies on RP and RMI links for failover and role management.
- **ARP table:** A database on a network device such as a switch that maintains mappings between IP addresses and MAC addresses; determines how to forward traffic within the local network.
(Analogy: Like a city map book held by delivery drivers in neighboring towns, showing them which mayor is at which city hall, so the right deliveries go to the right address.)
- **GARP (Gratuitous ARP):** A type of ARP (Address Resolution Protocol) packet broadcast by a controller, typically after a switchover, to update the ARP tables in connected network switches with the correct IP-to-MAC address mappings.
(Analogy: Like sending out an urgent memo to all nearby towns, telling them who the new mayor is so they update their city maps immediately after a leadership change.)
- **ARP cache timeout:** The duration for which an entry in the ARP table is considered valid before it must be refreshed; reducing this value helps the network recover quickly after role or address changes in the controllers.
(Analogy: Like setting a regular schedule for when all delivery drivers check for updates to the city map books, ensuring they quickly recognize changes in city leadership or addresses.)
- **SGACL (Security Group Access Control List):** A policy configuration that determines which types of network traffic (e.g., ICMP, ARP) are permitted between specific interfaces or devices, such as the RMI addresses of controllers.
(Analogy: Like special city rules that decide which types of emergency vehicles are allowed to travel the emergency road between city halls.)
- **ICMP (Internet Control Message Protocol):** A network protocol used for sending error messages and operational information between network devices; essential for controllers to monitor each other's status over the RMI.
(Analogy: Like sending quick bike couriers to report on the status of the roads or to alert if there's trouble between cities.)
- **AAA (Authentication, Authorization, and Accounting):** A framework for controlling user access to network resources and tracking user activity; controllers may send AAA-related packets from either the WMI IP or the RMI IP, so the AAA server must recognize both as valid.
(Analogy: Like having a security checkpoint that logs who enters or exits city buildings, whether they come from Main Street (WMI) or the emergency road (RMI).)
- **SGT (Security Group Tag):** An identification label assigned to network devices for policy enforcement with Cisco TrustSec; mapping is applied for both RMI and WMI addresses when device SGTs are used.
Analogy: Like giving every city vehicle a badge, so officials can enforce special rules for each group whether they use Main Street or the emergency road.
- **Cisco TrustSec:** A security architecture that uses SGTs to enforce network segmentation and access control policies; not supported on the RMI interface.
Analogy: Like a city's security zone system—effective on main city roads, but not supported on the emergency side road.

Limitations for RMI

- Cisco TrustSec is not supported on the RMI.

Best practices for RMI

- Ensure that the SGACL is defined appropriately to allow ICMP and ARP traffic between the active and standby RMI addresses when device SGT is used, since the IP-SGT mapping applies to both the RMI and WMI addresses.
- RMI-based "RP" High Availability is mandatory in the Cisco Catalyst CW9800H1 Wireless Controller, Cisco Catalyst CW9800H2 Wireless Controller and Cisco Catalyst CW9800M Wireless Controller.

Important: gateway monitoring interval and detection time

- When gateway reachability is enabled, both the active and standby controllers check gateway status through the RMI interface.
- It takes approximately the configured gateway monitoring interval to detect when a controller has lost gateway reachability.
- The default gateway monitoring interval is eight seconds, so the minimum detection time is about eight seconds unless you configure a different value.

Analogy

Think of redundancy management interfaces (RMI) as an emergency backup road that connects two cities (the active and standby controllers). The main highway (the redundancy port, RP) handles most of the day-to-day traffic and coordination. If the main highway is blocked or damaged, the emergency road (RMI) ensures that vital information—such as each city's health, status, and road conditions—can still be exchanged quickly.

Just as emergency vehicles and communication must be allowed to travel the backup road (analogous to allowing ICMP and ARP traffic via SGACL), the RMI helps both cities detect if both try to take charge at the same time (dual-active condition) and exchange important information about the area's main gateway (gateway status). If both the highway and the emergency road are inaccessible, each city might assume it's in charge, resulting in confusion (an IP conflict). The ARP table is like city maps in other towns (switches) that may not immediately recognize which city is currently in charge after both reconnect—setting these maps to update frequently (low ARP cache timeout) allows for faster recovery from major outages.

Controller operations with Redundant Management Interface

A controller operation with redundant management interface is a high-availability configuration that

- assigns the management IP address as the primary address on the active controller
- uses the secondary IPv4 address on the management VLAN as the RMI IP address for the active controller, and
- configures the RMI IP address as the primary IP address on the standby controller.

IP address assignment in controller operations

The active controller assigns IP addresses as follows:

- The primary address is the management IP address.
- The secondary IPv4 address on the management VLAN is the RMI (Redundant Management Interface) IP address for the active controller.

The standby controller manages IP addresses in a high-availability setup as follows:

- It does not have the wireless management IP configured.
- The RMI IP address is configured as the primary IP address on the standby controller.
- When the standby controller becomes active, the management IP address becomes the primary IP address, and the RMI IP address becomes the secondary IP address.
- If the interface on the active controller is administratively down, the same state is reflected on the standby controller.



Note Do not configure the secondary IPv4 address explicitly. RMI automatically configures a single secondary IPv4 address under the RMI.

Dual stack configurations on management vlans with RMI

A dual stack configuration is a network interface setup that

- allows both IPv4 and IPv6 addresses to be configured on the wireless management interface,
- permits monitoring only of the gateway that matches the configured RMI address family (IPv4 or IPv6), and
- restricts the visibility of the alternate family's management address on the standby controller.

Expanded explanation

- Dual stack refers to the fact that the wireless management interface can be configured with IPv4 and IPv6 addresses. If an RMI IPv4 address is configured along with an IPv4 management IP address, you can additionally configure an IPv6 management address on the wireless management interface. This IPv6 management IP address will not be visible on the standby controller.

-
-



Note The RMI feature supports only RMI IPv4 addresses.

RMI-based high-availability pairings

A RMI-based high-availability pair is a controller deployment configuration that

- uses Remote Machine Interface (RMI) to synchronize two controllers,

- provides redundancy by designating active and standby roles, and
- ensures failover and persistent state during controller reloads or outages.

RMI-based high-availability pairing scenarios and device support

You should consider RMI-based high-availability pairs in the following scenarios:

- Fresh installation: Configure high availability during the initial setup of controllers.
- Already paired controllers: Adjust or reconfigure pairing for controllers that are already part of a high-availability pair.
- Upgrade scenario: Maintain or update the pair relationship during software or hardware upgrades.
- Downgrade scenario: Ensure pairing remains stable and functional during downgrades.

Dynamic high-availability (HA) pairing requires both the active and standby controllers to reload. In practice, on the Cisco Catalyst 9800-L, 9800-40, and 9800-80 Wireless Controllers, dynamic pairing occurs when one controller reloads and becomes the standby member of the pair.



Note Unique chassis numbers must be configured for each controller before forming an HA pair, as these numbers identify the controllers within the pair.

HA pairing without previous configuration

A high-availability (HA) pairing without previous configuration is a deployment scenario for wireless controllers that

- initiates the HA setup on devices without existing ROMMON variables for RP (Route Processor) IP addresses
- allows selection between the soon-to-be-deprecated privileged EXEC mode RP-based commands and the newer RMI IP-based mechanisms, and
- derives RP IPs from RMI IPs after forming the HA pair, with restrictions on method transitions.

Command usage and method selection

When HA pairing is performed for the first time (without previous setup), devices do not have ROMMON variables for RP IP addresses.

After RMI-based HA pairing on a brand-new system:

- RP IPs are derived from RMI IPs and used in HA pairing.
- Privileged EXEC mode RP-based CLIs method of clearing and forming an HA pair is not allowed.
- To view the ROMMON variables, use the **show romvars** command.

Method selection considerations:

- You can still choose from the existing privileged EXEC mode RP-based commands or the RMI IP-based mechanisms. However, the privileged EXEC mode RP-based commands are deprecated.
- If you use Cisco Catalyst Center, you can choose the privileged EXEC mode RP-based CLI mechanism till the Cisco Catalyst Center migrates to support the RMI.
- If you choose privileged EXEC RP-based CLI mechanism, the RP IPs are configured the same way as in the 16.12 release.

Use the RMI IP-based mechanism for fresh installations, even though both RP-based and RMI methods may initially be available.

Software version requirements:

- The RMI migration is supported from Cisco Catalyst Center, 2.3.3.x release version.
- RMI-based High Availability requires Cisco IOS XE release version 17.3 or above.

Cisco Catalyst Center interoperability:

- If you use Cisco Catalyst Center, you can choose the privileged EXEC mode RP-based CLI mechanism till the Cisco Catalyst Center migrates to support the RMI.
- The RMI migration is supported from Cisco Catalyst Center, 2.3.3.x release version.
- Devices are not reachable.
- Non-Cisco Catalyst 9800 Series Wireless Controllers are in use.
- Controller is running Cisco IOS XE 17.3 or below
- High Availability is not configured.
- High Availability RMI is already configured.
- Attempting upgrade to an already failed High Availability paired controller.

Paired controllers

A paired controller is a high availability (HA) infrastructure configuration that

- links two controllers to operate jointly for redundancy and failover,
- allows seamless migration from traditional EXEC mode RP-based commands to RMI-based HA pairing, and
- ensures controller identity and connectivity are maintained even when core pairing mechanisms are updated or reloaded.

Expanded explanation

When controllers are already in an HA pair, they continue to use existing EXEC mode RP-based commands unless Remote Management Interface (RMI) is enabled. Enabling RMI migrates the system to use RMI-derived HA pairing, overwriting any existing RP IPs with those derived from the RMI configuration. The HA pair remains stable immediately after this change, but the controllers only adopt the new IPs following their next reload.

RMI requires controllers to be reloaded for the changes to take effect. Once both controllers restart, they reestablish the HA pair using the new RMI-derived RP IPs. After pairing through RMI, EXEC mode RP-based commands are blocked, preventing configuration conflicts.

Examples

- Two controllers configured as a high availability pair, where enabling RMI changes the way their active-standby relationship is managed and what IPs are used for internal communication.
- An active and standby controller pair that continues functioning during migration from legacy RP-IP pairing to RMI, without disruption until reload.

Counter-examples

- Two standalone controllers operating independently without HA pairing cannot be considered paired controllers.
- A controller pair where RMI is never configured and all management remains through EXEC mode RP-based commands does not benefit from RMI-derived features.

Analogy

Imagine a paired controller setup as two co-pilots flying an airplane together (the airplane represents your network environment). Traditionally, they use walkie-talkies (EXEC mode RP-based commands) to coordinate their flying activities. If you upgrade their communication system to headsets (RMI-based pairing), the co-pilots continue flying the plane using walkie-talkies until they both put on the new headsets (after a "reload" or restart).

From that point onward, all their coordination happens via the more reliable and advanced headsets. The co-pilots' ability to work together—their partnership—remains unbroken throughout; it is only how they communicate and identify each other's messages that changes, and only becomes effective after both are using the new headsets.

Upgrade from Cisco IOS XE 16.1.x to a later release

When upgrading a system, you have these options:

- Migrate while retaining the existing RP IP configuration: In this scenario, the current RP IP configuration remains unchanged, and future modifications will utilize EXEC mode RP-based commands.
- Migrate after clearing the HA configuration: Here, you have the choice to use either the traditional EXEC mode RP-based commands or adopt the new RMI-based RP configuration. If the previous configuration is preserved, RMI will update the RP IPs with those derived from the RMI IPs.

Downgrade scenario



Important The downgrade scenario given below is not applicable for Cisco IOS XE 17.1.x.

In a downgrade scenario, only EXEC mode RP-based commands are available. The downgrade process may follow one of these paths:

- If the upgraded system used the RMI-based RP configuration.
- If the upgraded system continued to use the EXEC mode RP-based commands.

In the above cases, the downgraded system uses the EXEC mode RP-based commands to modify the configuration. However, the downgraded system will continue to use the new derived RP IPs.

In both of these cases, the system will revert to EXEC mode RP-based commands for configuration alterations, yet will still utilize the newly derived RP IPs.



Note When you downgrade the Cisco Catalyst 9800 Series Wireless Controller to any version below Cisco IOS XE 17.1 and if the mDNS gateway is enabled on the WLAN/RLAN/GLAN interfaces, the mdns-sd-interface gateway goes down after the downgrade.

To enable the mDNS gateway on the WLAN/RLAN/GLAN interfaces in Cisco IOS XE 16.12 and earlier versions, use these commands:

```
wlan test 1 test
```

```
mdns-sd gateway
```

To enable the mDNS gateway on the WLAN/RLAN/GLAN interfaces from version Cisco IOS XE 17.1 onwards, use these command:

```
mdns-sd-interface gateway
```

Configure redundancy management interface (GUI)

Enable redundancy management for Cisco Catalyst 9800 Series Wireless Controllers using the graphical user interface (GUI).

Use this task to configure the redundancy management interface (RMI) and set up either RMI+RP or RP redundancy pairing on Cisco Catalyst 9800 Series Wireless Controllers. Configuring redundancy improves system availability and failover capabilities.

Before you begin

Ensure that Wireless Management Interface (WMI) is available before configuring RMI + RP using the GUI.

Follow these steps to configure redundancy management interface using GUI:

Procedure

- Step 1** In the **Administration > Device > Redundancy** window, perform the following:
- a. Set the **Redundancy Configuration** toggle button to **Enabled** to activate redundancy configuration.
 - b. In the **Redundancy Pairing Type** field, select **RMI+RP** to perform RMI+RP redundancy pairing as follows:
 - In the **RMI IP for Chassis 1** field, enter the RMI IP address for chassis 1.

- In the **RMI IP for Chassis 2** field, enter the RMI IP address for chassis 2.
- c. In the **Redundancy Pairing Type** field, select **RP** to perform RP redundancy pairing as follows:
- In the **Local IP** field, enter an IP address for the local chassis.
 - In the **Netmask** field, enter the subnet mask assigned to all wireless clients.
 - From the **HA Interface** drop-down list, choose one of the HA interfaces.
You can select the HA interface only for Cisco Catalyst 9800 Series Wireless Controllers.
 - In the **Remote IP** field, enter an IP address for the remote chassis.
- d. In the **Keep Alive Timer** field, enter an appropriate timer value (1–10 ×100 milliseconds).
- e. In the **Keep Alive Retries** field, enter an appropriate retry value (3–10 seconds).
- f. In the **Active Chassis Priority** field, enter a value.

Step 2 Click **Apply** and reload controllers.

The redundancy management interface is configured, and redundancy pairing is established based on your chosen method. The controller is now set up for improved high availability and failover.

Configure redundancy management interface (CLI)

Configure a Redundancy Management Interface (RMI) on Cisco Catalyst 9800 controllers using CLI commands to support high availability (HA) between two devices.

Use this task when you want to set up high availability and redundancy between two Catalyst 9800 series controllers. The RMI coordinates HA communication and failover, ensuring service continuity in case of device failure.

Before you begin

- Ensure both controllers are cabled and powered on.
- Verify you have administrator access to both devices via CLI.
- Gather the following information:
 - Chassis number (1 or 2 for each controller)
 - Desired chassis priority for HA (if overriding default)
 - A dedicated GigabitEthernet interface for HA communication (required for 9800-CL controllers)
 - Management VLAN and corresponding IP addresses for each chassis

Procedure

Step 1 (Optional) Configure the priority of the specified device.

Example:

```
Device# chassis chassis-num priority chassis-priority
```

Example:

```
Device# chassis 1 priority 1
```

From Cisco IOS XE Gibraltar 16.12.x onwards, device reload is not required for the chassis priority to become effective.

- *chassis-num*—Enter the chassis number. The range is from 1 to 2.
- *chassis-priority*—Enter the chassis priority. The range is from 1 to 2. The default value is 1.

When both the devices boot up at the same time, the device with higher priority becomes active, and the other one becomes standby. If both the devices are configured with the same priority value, the one with the smaller MAC address acts as active and its peer acts as standby.

Step 2 Create an HA interface for your controller.

Example:

```
Device# chassis redundancy ha-interface GigabitEthernet interface-number
```

Example:

```
Device# chassis redundancy ha-interface  
GigabitEthernet 3
```

- *interface-number*: GigabitEthernet interface number. The range is from 1 to 32.

This step is applicable only for Cisco Catalyst 9800-CL Series Wireless Controllers. The chosen interface is used as the dedicated interface for HA communication between the 2 controllers.

Step 3 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 4 Configure Redundancy Management Interface.

Example:

```
Device(config)# redun-management interface vlan vlan-interface-number chassis chassis-number  
address ip-address chassis chassis-number address ip-address
```

Example:

```
Device(config)# redun-management interface  
Vlan 200 chassis 1 address 9.10.90.147  
chassis 2 address 9.10.90.149
```

- *vlan-interface-number*: VLAN interface number. The valid range is from 1 to 4094.
Here, the *vlan-interface-number* is the same VLAN as the Management VLAN. That is, both must be on the same subnet.
- *chassis-number*: Chassis number. The valid range is from 1 to 2.
- *ip-address*: Redundancy Management Interface IP address.

Each controller must have a unique chassis number for RMI to form the HA pair. The chassis number can be observed as SWITCH_NUMBER in the output of **show romvar** command. Modification of SWITCH_NUMBER is currently not available through the web UI.

To disable the HA pair, use the **no redun-management interface vlan chassis** command.

Step 5 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

Step 6 Save the configuration.

Example:

```
Device# write memory
```

Step 7 Reload the controllers.

Example:

```
Device# reload
```

When the RMI configuration is done, you must reload the controllers for the configuration to take effect.

For Cisco Catalyst 9800-CL Wireless Controller VM, both the active and standby controllers reload automatically. In the case of hardware platforms, you should reload the active controller manually, as only standby the controller reloads automatically.

The redundancy management interface is configured. After reload, an HA pair is established between the two controllers, enabling redundancy and failover support.