



## Service-Level High Availability

---

This chapter describes the Cisco NX-OS service restartability for service-level HA and includes the following sections:

- [Information About Cisco NX-OS Service Restarts, on page 1](#)
- [Restartability Infrastructure, on page 2](#)
- [Process Restartability, on page 3](#)
- [Restarts on Standby Supervisor Services, on page 5](#)
- [Restarts on Switching Module Services, on page 5](#)
- [Restarts on Services Within a VDC, on page 5](#)
- [Troubleshooting Restarts, on page 6](#)
- [Related Documents, on page 6](#)
- [Standards, on page 7](#)
- [MIBs, on page 7](#)
- [RFCs, on page 7](#)
- [Technical Assistance, on page 7](#)

### Information About Cisco NX-OS Service Restarts

The Cisco NX-OS service restart features allows you to restart a faulty service without restarting the supervisor to prevent process-level failures from causing system-level failures. You can restart a service depending on current errors, failure circumstances, and the high-availability policy for the service. A service can undergo either a stateful or stateless restart. Cisco NX-OS allows services to store run-time state information and messages for a stateful restart. In a stateful restart, the service can retrieve this stored state information and resume operations from the last checkpoint service state. In a stateless restart, the service can initialize and run as if it had just been started with no prior state.

Not all services are designed for stateful restart. For example, Cisco NX-OS does not store run-time state information for Layer 3 routing protocols (such as Open Shortest Path First [OSPF] and Routing Information Protocol [RIP]). Their configuration settings are preserved across a restart, but these protocols are designed to rebuild their operational state using information obtained from neighbor routers. For details on the high-availability functionality of Layer 3 protocols, see chapter, *Network-Level High Availability* .

## Virtualization Support

For complete information on VDCs, see the *Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide*.

## Restartability Infrastructure

Cisco NX-OS allows stateful restarts of most processes and services. Back-end management and orchestration of processes, services, and applications within a platform are handled by a set of high-level system-control services.

## System Manager

The System Manager directs overall system function, service management, and system health monitoring, and enforces high-availability policies. The System Manager is responsible for launching, stopping, monitoring, restarting services, and initiating and managing the synchronization of service states and supervisor states for stateful switchover.

## Persistent Storage Service

Cisco NX-OS services use the persistent storage service (PSS) to store and manage the operational run-time information. The PSS component works with system services to recover states in the event of a service restart. PSS functions as a database of state and run-time information, which allows services to make a checkpoint of their state information whenever needed. A restarting service can recover the last known operating state that preceded a failure, which allows for a stateful restart.

Each service that uses PSS can define its stored information as private (it can be read only by that service) or shared (the information can be read by other services). If the information is shared, the service can specify that it is local (the information can be read only by services on the same supervisor) or global (it can be read by services on either supervisor or on modules). For example, if the PSS information of a service is defined as shared and global, services on other modules can synchronize with the PSS information of the service that runs on the active supervisor.

## Message and Transaction Service

The message and transaction service (MTS) is a high-performance interprocess communications (IPC) message broker that specializes in high-availability semantics. MTS handles message routing and queuing between services on and across modules and between supervisors. MTS facilitates the exchange of messages such as event notification, synchronization, and message persistency between system services and system components. MTS can maintain persistent messages and logged messages in queues for access even after a service restart.

MTS provides extensive serviceability features. For example, MTS provides notifications to inform an application when its queue has reached a predefined limitation. Corresponding to each notification, a default callback action is defined in MTS. From Cisco NX-OS Release 8.0(1), the System Message Logging contains new logs that indicates the highest MTS memory users. These logs are set to the severity level 4. Additionally, detailed memory usage stats with timestamps are collected per application. If an application contains an issue, you can use the command **show sys int mts sup sap APP\_SAP\_NUM queue\_stats** to collect the tech support.

```
MTS-4-MTS_MEM_USAGE: AT 886890 usecs after 11/16/2016 15:19:27: Total mem used: 76048384 (45%)
```

```
MTS-4-MTS_MEM_USAGE: top user 1: Node=0x101 vdc=1 sap=1 (SRVDESC_MTS_SYNC_THREAD)
bytes=77724160 msgs=9421
MTS-4-MTS_MEM_USAGE: top user 2: Node=0x109 vdc=3 sap=63 (mping_server) bytes=1736704 msgs=53
```

Explanation: MTS total memory usage and top memory users. This information will assist in debugging application issues.

Recommended Action: None.

## HA Policies

Cisco NX-OS allows each service to have an associated set of internal HA policies that define how a failed service will be restarted. Each service can have four defined policies—a primary and secondary policy when two supervisors are present, and a primary and secondary policy when only one supervisor is present. If no HA policy is defined for a service, the default HA policy to be performed upon a service failure will be a switchover if two supervisors are present, or a supervisor reset if only one supervisor is present.

Each HA policy specifies three parameters:

- Action to be performed by the System Manager:
  - Stateful restart
  - Stateless restart
  - Supervisor switchover (or restart)
- Maximum retries—Specifies the number of restart attempts to be performed by the System Manager. If the service has not restarted successfully after this number of attempts, the HA policy is considered to have failed, and the next HA policy is used. If no other HA policy exists, the default policy is applied, resulting in a supervisor switchover or restart.
- Minimum lifetime—Specifies the time that a service must run after a restart attempt to consider the restart attempt as successful. The minimum lifetime is no less than four minutes.

## Process Restartability

Process restartability ensures that a failed service can recover and resume operations without disrupting the data plane or other services. Depending on the service HA policies, previous restart failures, and the health of other services running on the same supervisor, the System Manager determines the action to be taken when a service fails.

The action taken by the System Manager for various failure conditions is described in the following table:

**Table 1: System Manager Action for Various Failure Cases**

Failure	
Service/process exception	Service restart
Service/process crash	Service restart
Unresponsive service/process	Service restart

Failure	
Repeated service failure	Supervisor reset (single) or switchover (dual)
Unresponsive System Manager	Supervisor reset (single) or switchover (dual)
Supervisor hardware failure	Supervisor reset (single) or switchover (dual)
Kernel failure	Supervisor reset (single) or switchover (dual)
Watchdog timeout	Supervisor reset (single) or switchover (dual)

## Types of Process Restarts

A failed service is restarted by one of the methods described in this section, depending on the service's HA implementation and HA policies,

### Stateful Restarts

When a restartable service fails, it is restarted on the same supervisor. If the new instance of the service determines that the previous instance was abnormally terminated by the operating system, the service then determines whether a persistent context exists. The initialization of the new instance attempts to read the persistent context to build a run-time context that makes the new instance appear like the previous one. After the initialization is complete, the service resumes the tasks that it was performing when it stopped. During the restart and initialization of the new instance, other services are unaware of the service failure. Any messages that are sent by other services to the failed service are available from the MTS when the service resumes.

Whether or not the new instance survives the stateful initialization depends on the cause of failure of the previous instance. If the service is unable to survive a few subsequent restart attempts, the restart is considered as failed. In this case, the System Manager performs the action specified by the HA policy of the services, forcing either a stateless restart, no restart, or a supervisor switchover or reset.

During a successful stateful restart, there is no delay while the system reaches a consistent state. Stateful restarts reduce the system recovery time after a failure.

The events before, during, and after a stateful restart are as follows:

1. The running services make a checkpoint of their run-time state information to the PSS.
2. The System Manager monitors the health of the running services that use heatbeats.
3. The System Manager restarts a service instantly when it crashes or hangs.
4. After restarting, the service recovers its state information from the PSS and resumes all pending transactions.
5. If the service does not resume a stable operation after multiple restarts, the System Manager initiates a reset or switchover of the supervisor.
6. Cisco NX-OS collects the process stack and core for debugging purposes with an option to transfer core files to a remote location.

When a stateful restart occurs, Cisco NX-OS sends a syslog message of level LOG\_ERR. If SNMP traps are enabled, the SNMP agent sends a trap. If the Smart Call Home service is enabled, the service sends an event message.

## Stateless Restarts

Cisco NX-OS infrastructure components manage stateless restarts. During a stateless restart, the System Manager identifies the failed process and replaces it with a new process. The service that failed does not maintain its run-time state upon the restart. The service can either build the run-time state from the running configuration, or if necessary, exchange information with other services to build a run-time state.

When a stateless restart occurs, Cisco NX-OS sends a syslog message of level LOG\_ERR. If SNMP traps are enabled, the SNMP agent sends a trap. If the Smart Call Home service is enabled, the service sends an event message.

## Switchovers

If a standby supervisor is available, Cisco NX-OS performs a supervisor switchover rather than a supervisor restart whenever multiple failures occur at the same time, because these cases are considered unrecoverable on the same supervisor. For example, if more than one HA application fails, that is considered an unrecoverable failure. For detailed information about supervisor switchovers and restarts, see chapter, *System level high availability*.

## Restarts on Standby Supervisor Services

When a service fails on a supervisor that is in the standby state, the System Manager does not apply the HA policies and restarts the service after a delay of 30 seconds. The delay ensures that the active supervisor is not overloaded by repeated standby service failures and synchronizations. If the service being restarted requires synchronization with a service on the active supervisor, the standby supervisor is taken out of hot standby mode until the service is restarted and synchronized. Services that are not restartable cause the standby supervisor to reset.

When a standby service restart occurs, Cisco NX-OS sends a syslog message of level LOG\_ERR. If SNMP traps are enabled, the SNMP agent sends a trap. If the Smart Call Home service is enabled, the service sends an event message.

## Restarts on Switching Module Services

When services fail on a switching module or another nonsupervisor module, the recovery action is determined by HA policies for those services. Because service failures on nonsupervisor module services do not require a supervisor switchover, the recovery options are a stateful restart, a stateless restart, or a module reset. If the module is capable of a nondisruptive upgrade, it is also capable of a nondisruptive restart.

When a nonsupervisor module service restart occurs, Cisco NX-OS sends a syslog message of level LOG\_ERR. If SNMP traps are enabled, the SNMP agent sends a trap. If the Smart Call Home service is enabled, the service sends an event message.

## Restarts on Services Within a VDC

When a service fails and all HA policies have been unsuccessful in restarting the service, the next action is typically a supervisor restart or switchover. However, if the service is running within a VDC, a VDC policy can specify that a restart of the VDC will be attempted before a supervisor restart or switchover.

For more information on VDCs, see the *Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide*.

## Troubleshooting Restarts

When a service fails, the system generates information that can be used to determine the cause of the failure. The following sources of information are available:

- Every service restart generates a syslog message of level LOG\_ERR.
- If the Smart Call Home service is enabled, every service restart generates a Smart Call Home event.
- If SNMP traps are enabled, the SNMP agent sends a trap when a service is restarted,
- When a service failure occurs on a local module, you can view a log of the event by entering the **show processes log** command in that module. The process logs are persistent across supervisor switchovers and resets.
- When a service fails, a system core image file is generated. You can view recent core images by using the **show cores** command on the active supervisor. Core files are not persistent across supervisor switchovers and resets, but you can configure the system to export core files to an external server by using a file transfer utility such as the Trivial File Transfer Protocol (TFTP).
- CISCO-SYSTEM-MIB contains a table for cores (cseSwCoresTable).

For information on collecting and using the generated information relating to service failures, see the *Cisco Nexus 7000 Series NX-OS Troubleshooting Guide*.

## Related Documents

Related Topic	Document Title
Virtual device context (VDC)	<i>Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide</i>
Supervisor switchovers	<i>System-Level High Availability</i>
Troubleshooting	<i>Cisco Nexus 7000 Series NX-OS Troubleshooting Guide</i>
Cisco NX-OS fundamentals	<i>Cisco Nexus 7000 Series NX-OS Fundamentals Configuration Guide</i>
Licensing	<i>Cisco Nexus 7000 Series NX-OS Licensing Guide</i>

## Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

## MIBs

MIBs	MIBs Link
<ul style="list-style-type: none"> <li>• CISCO-SYSTEM-EXT-MIB: ciscoHaGroup, cseSwCoresTable, cseHaRestartNotify, cseShutDownNotify, cseFailSwCoreNotify, cseFailSwCoreNotifyExtended</li> <li>• CISCO-PROCESS-MIB</li> <li>• CISCO-RF-MIB</li> </ul>	To locate and download MIBs, go to the following URL: <a href="https://cfngng.cisco.com/mibs">https://cfngng.cisco.com/mibs</a>

## RFCs

RFCs	Title
No RFCs are supported by this feature	—

## Technical Assistance

Description	Link
Technical Assistance Center (TAC) home page, containing 30,000 pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	<a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a>



**Note** This chapter refers to processes and services interchangeably. A process is considered to be a running instance of a service.

