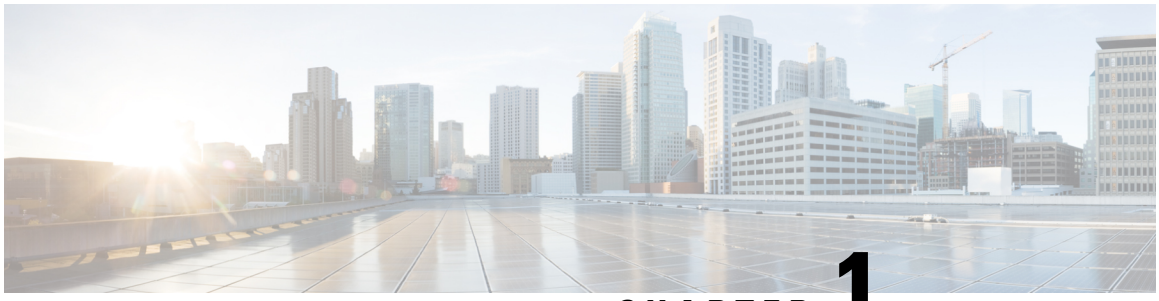
Cisco CSR 1000v and Cisco ISRv Software Configuration Guide

First Published: 2019-09-06

Last Modified: 2020-04-15

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS



CHAPTER 1

Preface

Preface

This preface describes the objectives and organization of this document and explains how to find additional information on related products and services. This preface contains the following sections:

Objectives

This document provides an overview of software functionality that is specific to the Cisco CSR 1000v Series Cloud Services Router. It is not intended as a comprehensive guide to all of the software features that can be run using the Cisco CSR 1000v Series router, but only the software aspects that are specific to this router.

For information on general software features that are also available on the Cisco CSR 1000v Series router, see the Cisco IOS XE technology guides for that specific software feature.

Related Documentation

This section refers you to other documentation that also might be useful as you configure your Cisco CSR 1000v router. The documentation listed below is available online. The following documents cover other important information for the Cisco CSR 1000v:

- Cisco CSR 1000V Documentation Roadmap, Cisco IOS XE Denali 16.x
- Cisco CSR 1000V Series Cloud Services Router Release Notes, Cisco IOS XE 3S
- Cisco CSR 1000V Series Cloud Services Router Release Notes for Cisco IOS XE Denali 16.x
- Cisco CSR 1000V Series Cloud Services Router Deployment Guide for Amazon Web Services
- Cisco CSR 1000v Deployment Guide for Microsoft Azure

- Cisco IOS XE REST API Management Reference Guide

The Cisco IOS XE release documentation home page contains technology guides and feature documentation:

See http://www.cisco.com/en/US/products/ps11174/tsd_products_support_series_home.html

For information on commands, see one of the following resources:

- Cisco IOS XE Software Command References
- Command Lookup Tool (cisco.com login required)

Document Conventions

This documentation uses the following conventions:

Convention	Description
^ or **Ctrl**	The ^ and **Ctrl** symbols represent the Control key. For example, the key combination **^D** or **Ctrl-D** means hold down the **Control** key while you press the **D** key. Keys are indicated in capital letters but are not case sensitive.
string	A string is a nonquoted set of characters shown in italics. For example, when setting an SNMP community string to public, do not use quotation marks around the string or the string will include the quotation marks.

Command syntax descriptions use the following conventions:

Convention	Description
bold	Bold text indicates commands and keywords that you enter exactly as shown.
italics	Italic text indicates arguments for which you supply values.
[x]	Square brackets enclose an optional element (keyword or argument).
|	A vertical line indicates a choice within an optional or required set of keywords or arguments.
[x | y]	Square brackets enclosing keywords or arguments separated by a vertical line indicate an optional choice.
{x | y}	Braces enclosing keywords or arguments separated by a vertical line indicate a required choice.

Nested sets of square brackets or braces indicate optional or required choices within optional or required elements. For example:

Convention	Description
[x {y | z}]	Braces and a vertical line within square brackets indicate a required choice within an optional element.

Examples use the following conventions:

Convention	Description
`screen`	Examples of information displayed on the screen are set in Courier font.
`bold screen`	Examples of text that you must enter are set in Courier bold font.
< >	Angle brackets enclose text that is not printed to the screen, such as passwords.
!	An exclamation point at the beginning of a line indicates a comment line. (Exclamation points are also displayed by the Cisco IOS XE software for certain processes.)
[ ]	Square brackets enclose default responses to system prompts.

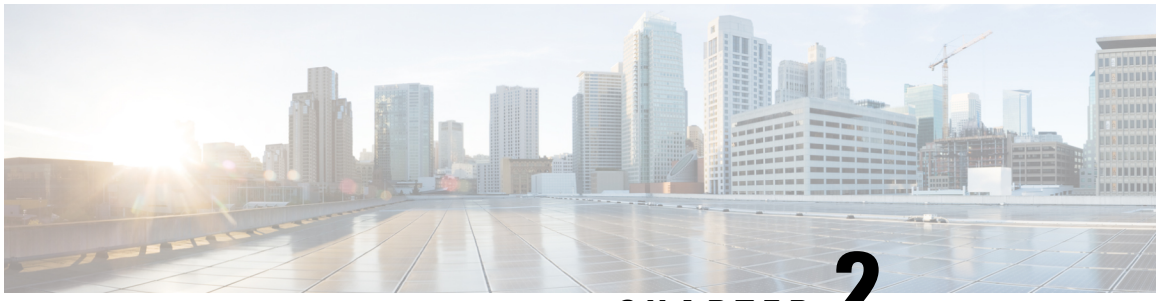
The following conventions are used to attract the attention of the reader:

Note Means *reader take note* . Notes contain helpful suggestions or references to materials that may not be contained in this manual.

Caution Means *reader be careful* . In this situation, you might do something that could result in equipment damage or loss of data.

Obtaining Documentation and Submitting a Service Request

For information about obtaining documentation, submitting a service request, and gathering additional information, see What's New in Cisco Product Documentation, which lists all new and revised Cisco technical documentation.



CHAPTER 2

Cisco CSR 1000v Series Cloud Services Router Overview

Introduction

Virtual Routers

The Cisco CSR 1000v Cloud Services Router provides a cloud-based virtual router deployed on a virtual machine (VM) instance on x86 server hardware. It supports a subset of Cisco IOS XE software features and technologies, providing Cisco IOS XE security and switching features on a virtualization platform.

The Cisco Integrated Services Virtual Router (Cisco ISRv) is very similar to the Cisco CSR 1000v. It provides a virtual IOS XE operating system for routing and forwarding on the Enterprise Network Compute System (ENCS) platform.

When the Cisco CSR 1000v is deployed on a VM, the Cisco IOS XE software functions just as if it were deployed on a traditional Cisco hardware platform.

Features

The Cisco CSR 1000v includes a virtual Route Processor and a virtual Forwarding Processor (FP) as part of its architecture. It supports a subset of Cisco IOS XE software features and technologies.

The Cisco CSR 1000v can provide secure connectivity from an enterprise location, such as a branch office or data center, to the public or private cloud.

The Cisco CSR 1000v is deployed as a virtual machine on a hypervisor. Optionally, you can use a virtual switch (vSwitch), depending on your deployment. You can use selected Cisco equipment for some components. The supported components will depend on your software release.

Benefits of Virtualization Using the Cisco CSR 1000v Series Cloud Services Router

The Cisco CSR 1000v Series uses the benefits of virtualization in the cloud to provide the following:

- Hardware independence

  Because the Cisco CSR 1000v runs on a virtual machine, it can be supported on any x86 hardware that the virtualization platform supports.

- Sharing of resources

  The resources used by the Cisco CSR 1000v are managed by the hypervisor, and resources can be shared among VMs. The amount of hardware resources that the VM server allocates to a specific VM can be reallocated to another VM on the server.

- Flexibility in deployment

  You can easily move a VM from one server to another. Thus, you can move the Cisco CSR 1000v from a server in one physical location to a server in another physical location without moving any hardware resources.

Software Configuration and Management Using the Cisco IOS XE CLI

You can perform software configuration and management of the Cisco CSR 1000v using the following methods:

- Provision a serial port in the VM and connect to access the Cisco IOS XE CLI commands.
- Use the virtual VGA console or the console on the virtual serial port to access the Cisco IOS XE CLI commands.

Note A serial port can be used to manage a Cisco CSR 1000v VM only if the underlying hypervisor supports associating a serial port with a VM. For example, the Citrix XenServer environment does not support serial port association. See your hypervisor documentation for details.

- Use remote SSH/Telnet to access the Cisco IOS XE CLI commands.

The Cisco CSR 1000v also supports management and configuration using the following products:

- Cisco IOS XE REST API
- Cisco Prime Network Services Controller

For more information, see "Management Support", from Managing the Router Using Cisco Configuration Professional, on page 50 onwards.

Router Interfaces

The Cisco CSR 1000v router interfaces perform the same functionality as those on hardware-based Cisco routers. The Cisco CSR 1000v interfaces function as follows:

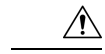
- Interfaces are logically named as the Gigabit Ethernet (GE) interfaces.
- The available interface numbering depends on the Cisco CSR 1000v version.

(Cisco IOS XE Release 3.11S and later, and Denali 16.2 and later) The interface numbering is as follows:

- - Interface port numbering is from 1 and up to the number of interfaces supported.
  - GigabitEthernet interface 0 is no longer supported beginning with this release.
  - You can designate any interface as the management interface. You can change the management interface when deploying the OVA template on first-time installation.

(Cisco IOS XE Release 3.10S and earlier) The interface numbering is as follows:

- - Interface port numbering is from 0 and up to the number of interfaces supported.
  - Gigabit Ethernet interface 0 is reserved for the management interface used for obtaining the licenses and upgrading software.
- At first boot, the Cisco CSR 1000v router interfaces are mapped to the vNIC interfaces on the VM based on the vNIC enumeration to the Cisco CSR 1000v; on subsequent boot, the Cisco CSR 1000v router interfaces are mapped to the vNIC MAC address



Caution If upgrading to Cisco IOS XE Release 3.11S from an earlier release, we recommend you update your configuration to remove the GigabitEthernet 0 management interface before upgrading. Because the GigabitEthernet 0 interface is no longer supported beginning with Cisco IOS XE Release 3.11S, you will receive system errors if the upgraded configuration includes this interface.

For more information, see the "Mapping Cisco CSR 1000v Network Interfaces to VM Network Interfaces" section on page 11-1 .

Virtual Machine Requirements

The Cisco CSR 1000v runs only on a virtual machine. This section describes the virtual machine requirements for the router.

- Virtual Machines, on page 7
- Hypervisor Support, on page 8
- Server Requirements, on page 33

Virtual Machines

A virtual machine (VM) is a software implementation of a computing environment in which an operating system (OS) or program can be installed and run. The VM typically emulates a physical computing environment, but requests for CPU, memory, hard disk, network and other hardware resources are managed by a virtualization layer which translates these requests to the underlying physical hardware.

You can deploy an Open Virtualization Archive (OVA) file. The OVA file package simplifies the process of deploying a VM by providing a complete definition of the parameters and resource allocation requirements for the new VM.

An OVA file consists of a descriptor (.ovf) file, a storage (.vmdk) file and a manifest (.mf) file.

- ovf file—Descriptor file which is an xml file with extension .ovf which consists of all the metadata about the package. It encodes all the product details, virtual hardware requirements and licensing.
- vmdk file—File format that encodes a single virtual disk from a VM.
- mf file—Optional file that stores the SHA key generated during packaging.

You can also install the Cisco CSR 1000v using an .iso file and manually create the VM in the hypervisor.

For more information, see the "Installation Overview" section on page 3-1 .

Hypervisor Support

A hypervisor enables multiple operating systems to share a single hardware host machine. While each operating system appears to have the dedicated use of the host's processor, memory, and other resources; the hypervisor controls and allocates only needed resources to each operating system and ensures that the operating systems (VMs) do not disrupt each other.

Supported Hypervisor Types

Installation of the Cisco CSR 1000v is supported on selected **Type 1** (native, bare metal) hypervisors. Installation is not supported on **Type 2** (hosted) hypervisors, such as VMware Fusion, VMware Player, or Virtual Box.

Amazon Cloud Marketplace

The Cisco CSR 1000v is available in the Amazon Cloud Marketplace. (For use with Cisco IOS XE Release 3.11S through 3.16.2S, and Cisco IOS XE Denali 16.3.1 and later.) For more information, see the Cisco CSR 1000V Series Cloud Services Router Deployment Guide for Amazon Web Services.

Microsoft Azure Marketplace

The Cisco CSR 1000v is available in the Microsoft Azure Marketplace . For more information, see the Cisco CSR 1000V Series Cloud Services Router Deployment Guide for Microsoft Azure .

Hypervisor Versions for Cisco IOS XE Denali 16.3.1 and Later

The following hypervisors/versions are supported by Cisco CSR 1000v on Cisco IOS XE Denali 16.3.1 and later. If you are using older versions of Cisco IOS XE, see Hypervisor Versions—Cisco IOS XE 3.x, on page 9.

VMware ESXi

Server 6.0 update2 (instance running vm11)—recommended. F.ully tested and meets performance benchmarks

Server 5.5 update3 (instance running vm10)

Although 5.5 update 3 is supported for Cisco IOS XE Denali 16.3.1 and later, we recommend using VMware ESXi Server 6.0 update 2 instead.

Kernel Based Virtual Machine (KVM)

RHEL 7.2—recommended

RHEL 7.1

Citrix XenServer

6.5—recommended

6.2

Microsoft Hyper-V

Windows Server 2012-R2, Hyper-V Mgr 6.3.9600.16384—recommended

Amazon Web Services

C4 and T2 instance types—recommended

C3 instance types—supported (in Cisco IOS XE 3.11 to IOS XE 3.17)

See "Amazon Web Services" in Hypervisor Support, on page 8.

Microsoft Azure

Standard D2 and Standard D3—recommended

Standard D4—supported

See "Microsoft Azure" in Hypervisor Support, on page 8.

Hypervisor Versions—Cisco IOS XE 3.x

The following table lists the supported hypervisor versions for older software releases (Cisco IOS XE 3.x).

Note For recent hypervisor versions see the *Hypervisor Versions for Cisco IOS XE Denali 16.3.1 and Later* section.

Table 1: Support Matrix for Hypervisor Versions

Cisco CSR 1000v IOS XE Release	VMwareESXi	Citrix XenServer	Kernel Based Virtual Machine (KVM)	Microsoft Hyper-V	Amazon Web Services
3.9S	5.0	Not supported	Not supported	Not supported	Not supported
3.10S	5.05.1	6.0.2	• Linux KVM based on Red Hat Enterprise Linux 6.3[1] • Red Hat Enterprise Virtualization 3.1	Not supported	Not supported

Cisco CSR 1000v IOS XE Release	VMwareESXi	Citrix XenServer	Kernel Based Virtual Machine (KVM)	Microsoft Hyper-V	Amazon Web Services
3.11S	5.05.1	6.02	• Linux KVM based on Red Hat Enterprise Linux 6.31 • Red Hat Enterprise Virtualization 3.1 • Ubuntu 12.04.03 LTS Server 64 Bits[2]	Not supported	Supported
3.12S	5.05.15.5[3]	6.1	• Linux KVM based on Red Hat Enterprise Linux 6.31 • Ubuntu 12.04.03 LTS Server 64 Bits 2	Windows Server 2012 R2	Supported
3.13S	5.05.15.5[4]	6.2	• Linux KVM based on Red Hat Enterprise Linux 6.31 • Ubuntu 12.04.03 LTS Server 64 Bits 2	Windows Server 2012 R2	Supported
3.14S	5.05.15.5[5]	6.2	• Linux KVM based on Red Hat Enterprise Linux 6.5 • Ubuntu 14.04 LTS Server 64 Bits 2	Windows Server 2012 R2	Supported

Cisco CSR 1000v IOS XE Release	VMwareESXi	Citrix XenServer	Kernel Based Virtual Machine (KVM)	Microsoft Hyper-V	Amazon Web Services
3.15S	5.05.15.5[6]	6.2	• Linux KVM based on Red Hat Enterprise Linux 6.6 • Ubuntu 14.04 LTS Server 64 Bits 2	Windows Server 2012 R2	Supported
3.16S	5.05.15.5[7]6.0[8]	6.2	• Linux KVM based on Red Hat Enterprise Linux 6.6 • Ubuntu 14.04 LTS Server 64 Bits 2	Windows Server 2012 R2	Supported until Cisco IOS XE 3.16.2
3.17S	5.05.15.56.0	6.2	• Linux KVM based on Red Hat Enterprise Linux 7.1[9] • Ubuntu 14.04 LTS Server 64 Bits 2	Windows Server 2012 R2	Not supported
For later versions of Cisco IOS XE, see Hypervisor Versions for Cisco IOS XE Denali 16.3.1 and Later, on page 8					

[1] Requires Kernel version 2.6.3.2 and QEMU 0.12.
[2] Requires QEMU-x86_64 version 1.0 (qemu-kvm-1.0), Copyright (c) 2003-2008 Fabrice Bellard.
[3] VMware ESXi 5.5 update 3 is not supported on Cisco IOS XE 3.12S..
[4] VMware ESXi 5.5 update 3 is not supported on Cisco IOS XE 3.13S..
[5] VMware ESXi 5.5 update 3 is not supported on Cisco IOS XE 3.14S.
[6] VMware ESXi 5.5 update 3 is not supported on Cisco IOS XE 3.15S.
[7] VMware ESXi 5.5 update 3 is supported on Cisco IOS XE 3.16.1S and later.
[8] VMware ESXi 6.0 supported on Cisco IOS XE 3.16.1S and later (and 3.17S and later).
[9] Requires Kernel version 3.10.0 and QEMU 1.5.3.

Hypervisor features may differ depending on the hypervisor, and not all features in a given hypervisor version may be supported. The hypervisor versions listed are those officially tested and supported by the Cisco CSR 1000v. See the following sections for more information:

Note For information about deploying the Cisco CSR 1000v in an Amazon Web Services environment, see the

Note For information about deploying the Cisco CSR 1000v in a Microsoft Azure environment, see the Cisco CSR 1000v Deployment Guide for Microsoft Azure.

Hypervisor vNIC Requirements

Depending on the Cisco CSR 1000v release version, each of the hypervisors supports different virtual network interface card (vNIC) types. The Cisco CSR 1000v also supports a different maximum number of vNICs depending on the hypervisor. Some versions and hypervisors also support the ability to add and remove vNICs without powering down the VM (for example, vNIC Hot Add/Remove).

The VMXNET3, VIF and Virtio NIC types listed in the table are para-virtualized NICs.

See also Supported I/O Modes and Drivers, on page 28.

Note: PCI Passthrough: enic is not supported in Cisco IOS XE Denali 16.3.1 and higher.

The following sections list the supported vNICs and the minimum and maximum number of vNICs supported for each VM instance. Choose a section, depending on the release of Cisco IOS XE which you are using.

Hypervisor vNIC Requirements for Cisco IOS XE Gibraltar 17.1 Release

vNIC Requirements for VMware ESXi	Value
NIC Types Supported	VMXNET3 ixgbe(Intel 10Gb PCI Express NIC Driver), ixgbevf, i40evf
Max. number of vNICs per VM instance	10
vNIC Hot Add Support (Intel 10Gb PCI Express NIC Driver)	Yes
vNIC Hot Remove Support (Intel 10Gb PCI Express NIC Driver)	Yes
vNIC Two-Step Hot Remove Support (Intel 10Gb PCI Express NIC Driver)	Yes
Single Root I/O virtualization (SR-IOV) Support	Yes

vNIC Requirements for Citrix XenServer	Value
NIC Types Supported	VIF-netfront(pmap)
Max. number of vNICs per VM instance	7
vNIC Hot Add Support	Yes
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for KVM	Value
NIC Types Supported	Virtio, ixgbevf, ixgbe, i40evf
Max. number of vNICs per VM instance	26
vNIC Hot Add Support	Yes
vNIC Hot Remove Support	Yes
vNIC Two-Step Hot Remove Support	Yes
Single Root I/O virtualization (SR-IOV) Support	Yes

vNIC Requirements for Microsoft Hyper-V	Value
NIC Types Supported	NetVSC
Max. number of vNICs per VM instance	8
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for Microsoft Azure	Value
NIC Types Supported	NetVSC
Max. number of vNICs per VM instance	8
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for Amazon Web Services (AWS)	Value
NIC Types Supported	VIF-netfront(pmap), ixgbevf
Max. number of vNICs per VM instance	8 (See http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-eni.html#AvailableIpPerENI)
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

Hypervisor vNIC Requirements for Cisco IOS XE Gibraltar 16.10, 16.11 and 16.12 releases

vNIC Requirements for VMware ESXi	Value
NIC Types Supported	VMXNET3 ixgbe(Intel 10Gb PCI Express NIC Driver), ixgbevf, i40evf
Max. number of vNICs per VM instance	10
vNIC Hot Add Support (Intel 10Gb PCI Express NIC Driver)	Yes
vNIC Hot Remove Support (Intel 10Gb PCI Express NIC Driver)	No
vNIC Two-Step Hot Remove Support (Intel 10Gb PCI Express NIC Driver)	Yes
Single Root I/O virtualization (SR-IOV) Support	Yes

vNIC Requirements for Citrix XenServer	Value
NIC Types Supported	VIF-netfront(pmap)
Max. number of vNICs per VM instance	7
vNIC Hot Add Support	Yes
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for KVM	Value
NIC Types Supported	Virtio, ixgbevf, ixgbe, i40evf
Max. number of vNICs per VM instance	26
vNIC Hot Add Support	Yes
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	Yes
Single Root I/O virtualization (SR-IOV) Support	Yes

vNIC Requirements for Microsoft Hyper-V	Value
NIC Types Supported	NetVSC
Max. number of vNICs per VM instance	8
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for Microsoft Azure	Value
NIC Types Supported	NetVSC
Max. number of vNICs per VM instance	8
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for Amazon Web Services (AWS)	Value
NIC Types Supported	VIF-netfront(pmap), ixgbevf
Max. number of vNICs per VM instance	8 (See http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-eni.html#AvailableIpPerENI)
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

Hypervisor vNIC Requirements for Cisco IOS XE Fuji 16.9

vNIC Requirements for VMware ESXi	Value
NIC Types Supported	VMXNET3 ixgbe(Intel 10Gb PCI Express NIC Driver), ixgbevf, i40evf
Max. number of vNICs per VM instance	10
vNIC Hot Add Support (Intel 10Gb PCI Express NIC Driver)	Yes
vNIC Hot Remove Support (Intel 10Gb PCI Express NIC Driver)	No
vNIC Two-Step Hot Remove Support (Intel 10Gb PCI Express NIC Driver)	Yes
Single Root I/O virtualization (SR-IOV) Support	Yes

vNIC Requirements for Citrix XenServer	Value
NIC Types Supported	VIF-netfront(pmap)
Max. number of vNICs per VM instance	7
vNIC Hot Add Support	Yes
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	No

vNIC Requirements for Citrix XenServer	Value
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for KVM	Value
NIC Types Supported	Virtio, ixgbevf, ixgbe, i40evf
Max. number of vNICs per VM instance	26
vNIC Hot Add Support	Yes
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	Yes
Single Root I/O virtualization (SR-IOV) Support	Yes

vNIC Requirements for Microsoft Hyper-V	Value
NIC Types Supported	NetVSC
Max. number of vNICs per VM instance	8
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for Microsoft Azure	Value
NIC Types Supported	NetVSC
Max. number of vNICs per VM instance	8
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for Amazon Web Services (AWS)	Value
NIC Types Supported	VIF-netfront(pmap), ixgbevf
Max. number of vNICs per VM instance	8 (See http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-eni.html#AvailableIpPerENI)
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	No

vNIC Requirements for Amazon Web Services (AWS)	Value
Single Root I/O virtualization (SR-IOV) Support	No

Hypervisor vNIC Requirements for Cisco IOS XE Fuji 16.8

vNIC Requirements for VMware ESXi	Value
NIC Types Supported	VMXNET3 ixgbe(Intel 10Gb PCI Express NIC Driver), ixgbevf, i40evf
Max. number of vNICs per VM instance	10
vNIC Hot Add Support (Intel 10Gb PCI Express NIC Driver)	Yes
vNIC Hot Remove Support (Intel 10Gb PCI Express NIC Driver)	No
vNIC Two-Step Hot Remove Support (Intel 10Gb PCI Express NIC Driver)	Yes
Single Root I/O virtualization (SR-IOV) Support	Yes

vNIC Requirements for Citrix XenServer	Value
NIC Types Supported	VIF-netfront(pmap)
Max. number of vNICs per VM instance	7
vNIC Hot Add Support	Yes
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for KVM	Value
NIC Types Supported	Virtio, ixgbevf, ixgbe, i40evf
Max. number of vNICs per VM instance	26
vNIC Hot Add Support	Yes
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	Yes
Single Root I/O virtualization (SR-IOV) Support	Yes

vNIC Requirements for Microsoft Hyper-V	Value
NIC Types Supported	NetVSC
Max. number of vNICs per VM instance	8
vNIC Hot Add Support	No
vNIC Hot Remove Support	No

vNIC Requirements for Microsoft Hyper-V	Value
vNIC Two-Step Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for Microsoft Azure	Value
NIC Types Supported	NetVSC
Max. number of vNICs per VM instance	8
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for Amazon Web Services (AWS)	Value
NIC Types Supported	VIF-netfront(pmap), ixgbevf
Max. number of vNICs per VM instance	8 (See http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-eni.html#AvailableIpPerENI)
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

Hypervisor vNIC Requirements for Cisco IOS XE Fuji 16.7

vNIC Requirements for VMware ESXi	Value
NIC Types Supported	VMXNET3 ixgbe(Intel 10Gb PCI Express NIC Driver), ixgbevf, i40evf
Max. number of vNICs per VM instance	10
vNIC Hot Add Support (Intel 10Gb PCI Express NIC Driver)	Yes
vNIC Hot Remove Support (Intel 10Gb PCI Express NIC Driver)	No
vNIC Two-Step Hot Remove Support (Intel 10Gb PCI Express NIC Driver)	Yes
Single Root I/O virtualization (SR-IOV) Support	Yes

vNIC Requirements for Citrix XenServer	Value
NIC Types Supported	VIF-netfront(pmap)

vNIC Requirements for Citrix XenServer	Value
Max. number of vNICs per VM instance	7
vNIC Hot Add Support	Yes
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for KVM	Value
NIC Types Supported	Virtio, ixgbevf, ixgbe, i40evf
Max. number of vNICs per VM instance	26
vNIC Hot Add Support	Yes
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	Yes
Single Root I/O virtualization (SR-IOV) Support	Yes

vNIC Requirements for Microsoft Hyper-V	Value
NIC Types Supported	NetVSC
Max. number of vNICs per VM instance	8
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for Microsoft Azure	Value
NIC Types Supported	NetVSC
Max. number of vNICs per VM instance	8
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for Amazon Web Services (AWS)	Value
NIC Types Supported	VIF-netfront(pmap), ixgbevf

vNIC Requirements for Amazon Web Services (AWS)	Value
Max. number of vNICs per VM instance	8 (See http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-eni.html#AvailableIpPerENI)
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
vNIC Two-Step Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

Hypervisor vNIC Requirements for Cisco IOS XE Everest 16.6

vNIC Requirements for VMware ESXi	Value
NIC Types Supported	VMXNET3 ixgbe(Intel 10Gb PCI Express NIC Driver) ixgbevf
Max. number of vNICs per VM instance	10
vNIC Hot Add Support (Intel 10Gb PCI Express NIC Driver)	Yes
vNIC Hot Remove Support (Intel 10Gb PCI Express NIC Driver)	No
Single Root I/O virtualization (SR-IOV) Support	Yes

vNIC Requirements for Citrix XenServer	Value
NIC Types Supported	VIF-netfront(pmap)
Max. number of vNICs per VM instance	7
vNIC Hot Add Support	Yes
vNIC Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for KVM	Value
NIC Types Supported	Virtio, ixgbevf, ixgbe
Max. number of vNICs per VM instance	26
vNIC Hot Add Support	Yes
vNIC Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	Yes

vNIC Requirements for Microsoft Hyper-V	Value
NIC Types Supported	NetVSC
Max. number of vNICs per VM instance	8
vNIC Hot Add Support	No

vNIC Requirements for Microsoft Hyper-V	Value
vNIC Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for Microsoft Azure	Value
NIC Types Supported	NetVSC
Max. number of vNICs per VM instance	8
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for Amazon Web Services (AWS)	Value
NIC Types Supported	VIF-netfront(pmap), ixgbevf
Max. number of vNICs per VM instance	8 (See http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-eni.html#AvailableIpPerENI)
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

Hypervisor vNIC Requirements for Cisco IOS XE Everest 16.5

vNIC Requirements for VMware ESXi	Value
NIC Types Supported	VMXNET3 ixgbe(Intel 10Gb PCI Express NIC Driver) ixgbevf
Max. number of vNICs per VM instance	10
vNIC Hot Add Support (Intel 10Gb PCI Express NIC Driver)	Yes
vNIC Hot Remove Support (Intel 10Gb PCI Express NIC Driver)	No
Single Root I/O virtualization (SR-IOV) Support	Yes

vNIC Requirements for Citrix XenServer	Value
NIC Types Supported	VIF-netfront(pmap)
Max. number of vNICs per VM instance	7
vNIC Hot Add Support	Yes
vNIC Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for KVM	Value
NIC Types Supported	Virtio, ixgbevf, ixgbe
Max. number of vNICs per VM instance	26
vNIC Hot Add Support	Yes
vNIC Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	Yes

vNIC Requirements for Microsoft Hyper-V	Value
NIC Types Supported	NetVSC
Max. number of vNICs per VM instance	8
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for Microsoft Azure	Value
NIC Types Supported	NetVSC
Max. number of vNICs per VM instance	8
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for Amazon Web Services (AWS)	Value
NIC Types Supported	VIF-netfront(pmap), ixgbevf
Max. number of vNICs per VM instance	8 (See http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-eni.html#AvailableIpPerENI)
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

Hypervisor vNIC Requirements for Cisco IOS XE Everest 16.4

vNIC Requirements for VMware ESXi	Value
NIC Types Supported	VMXNET3 ixgbe(Intel 10Gb PCI Express NIC Driver) ixgbevf
Max. number of vNICs per VM instance	10
vNIC Hot Add Support (Intel 10Gb PCI Express NIC Driver)	Yes

vNIC Requirements for VMware ESXi	Value
vNIC Hot Remove Support (Intel 10Gb PCI Express NIC Driver)	No
Single Root I/O virtualization (SR-IOV) Support	Yes

vNIC Requirements for Citrix XenServer	Value
NIC Types Supported	VIF-netfront(pmap)
Max. number of vNICs per VM instance	7
vNIC Hot Add Support	Yes
vNIC Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for KVM	Value
NIC Types Supported	Virtio, ixgbevf, ixgbe
Max. number of vNICs per VM instance	26
vNIC Hot Add Support	Yes
vNIC Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	Yes

vNIC Requirements for Microsoft Hyper-V	Value
NIC Types Supported	NetVSC
Max. number of vNICs per VM instance	8
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for Microsoft Azure	Value
NIC Types Supported	NetVSC
Max. number of vNICs per VM instance	4
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for Amazon Web Services (AWS)	Value
NIC Types Supported	VIF-netfront(pmap), ixgbevf

vNIC Requirements for Amazon Web Services (AWS)	Value
Max. number of vNICs per VM instance	8 (See http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-eni.html#AvailableIpPerENI)
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

Hypervisor vNIC Requirements for Cisco IOS XE Denali 16.3

vNIC Requirements for VMware ESXi	Value
NIC Types Supported	VMXNET3 ixgbe (Intel 10Gb PCI Express NIC Driver) ixgbevf
Max. number of vNICs per VM instance	10
vNIC Hot Add Support (Intel 10Gb PCI Express NIC Driver)	Yes
vNIC Hot Remove Support (Intel 10Gb PCI Express NIC Driver)	No
Single Root I/O virtualization (SR-IOV) Support	Yes

vNIC Requirements for Citrix XenServer	Value
NIC Types Supported	VIF-netfront(pmap)
Max. number of vNICs per VM instance	7
vNIC Hot Add Support	Yes
vNIC Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for KVM	Value
NIC Types Supported	Virtio, ixgbevf, ixgbe
Max. number of vNICs per VM instance	26
vNIC Hot Add Support	Yes
vNIC Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	Yes

vNIC Requirements for Microsoft Hyper-V	Value
NIC Types Supported	NetVSC
Max. number of vNICs per VM instance	8
vNIC Hot Add Support	No
vNIC Hot Remove Support	No

vNIC Requirements for Microsoft Hyper-V	Value
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for Microsoft Azure	Value
NIC Types Supported	NetVSC
Max. number of vNICs per VM instance	4
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

vNIC Requirements for Amazon Web Services (AWS)	Value
NIC Types Supported	VIF-netfront(pmap), ixgbevf
Max. number of vNICs per VM instance	8 (See http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-eni.html#AvailableIpPerENI)
vNIC Hot Add Support	No
vNIC Hot Remove Support	No
Single Root I/O virtualization (SR-IOV) Support	No

Hypervisor vNIC Requirements for Cisco IOS XE 3S Releases

Table 2: Cisco CSR 1000v vNIC Support for Cisco IOS XE 3S Releases

Cisco IOS XE Release:	3.9S	3.10S, 3.11S	3.12S	3.13S, 3.14S, 3.15S	3.16S, 3.17S
VMware ESXi	VMXNET3	VMXNET3	VMXNET3	VMXNET3 ixgbe (Intel 10Gb PCI Express NIC Driver)	VMXNET3 ixgbe (Intel 10Gb PCI Express NIC Driver) enic
NIC Types Supported					
Max. number of vNICs per VM instance	10	10	10	10	10

Cisco IOS XE Release:	3.9S	3.10S, 3.11S	3.12S	3.13S, 3.14S, 3.15S	3.16S, 3.17S
vNIC Hot Add/Remove Support(Prior to release 3.15S, vNIC Hot Remove requires reloading the Cisco CSR 1000v. This is applicable only when using the VMXNET3 driver.)	No	Yes	Yes	Yes	Yes
Single Root I/O virtualization (SR-IOV) Support	—	—	No	No	No
Citrix XenServer					
NIC Types Supported	—	VIF	VIF	VIF ixgbevf ixgbe (Intel 10Gb PCI Express NIC Driver)	VIF ixgbevf ixgbe (Intel 10Gb PCI Express NIC Driver)
Max. number of vNICs per VM instance	—	7	7	7	7
vNIC Hot Add/Remove Support	—	No	No	No	No
Single Root I/O virtualization (SR-IOV) Support	—	—	Yes (from release 3.12.1S)	Yes	Yes
KVM					
NIC Types Supported	—	Virtio	Virtio	Virtio ixgbevf ixgbe (Intel 10Gb PCI Express NIC Driver)	Virtio ixgbevf ixgbe (Intel 10Gb PCI Express NIC Driver)enic
Max. number of vNICs per VM instance	—	10	26	26	26

Cisco IOS XE Release:	3.9S	3.10S, 3.11S	3.12S	3.13S, 3.14S, 3.15S	3.16S, 3.17S
vNIC Hot Add/Remove Support (Prior to release 3.15S, vNIC Hot Remove requires reloading the Cisco CSR 1000v. This is applicable only when using the Virtio driver)	—	Yes	Yes	Yes	Yes
Single Root I/O virtualization (SR-IOV) Support	—	—	Yes (from release 3.12.1S)	Yes (Requires the host hardware to support the Intel VT-d or AMD IOMMU specification. SR-IOV is not supported with Virtual LANs (VLANs))	Yes(Requires the host hardware to support the Intel VT-d or AMD IOMMU specification. SR-IOV is not supported with Virtual LANs (VLANs))
Microsoft Hyper-V					
NIC Types Supported	—	—	HV NetVSC	HV NetVSC	HV NetVSC
Max. number of vNICs per VM instance	—	—	3	3	3
vNIC Hot Add/Remove Support	—	—	No	No	No
Single Root I/O virtualization (SR-IOV) Support	—	—	No	No	No
Amazon Web Services					
NIC Types Supported	—	(For Cisco IOS XE 3.11 and later), aws-vif(pmap)	aws-vif(pmap)	aws-vif(pmap)	(Up until Cisco IOS XE 3.16.2) aws-vif(pmap), aws-ixgbevf(SRIOV)

Cisco IOS XE Release:	3.9S	3.10S, 3.11S	3.12S	3.13S, 3.14S, 3.15S	3.16S, 3.17S
Max. number of vNICs per VM instance	—	(For Cisco IOS XE 3.11 to 3.16.2, number depends on instance type. For more information, see the AWS User Guide.			
vNIC Hot Add/Remove Support	—	No	No	No	No
Single Root I/O virtualization (SR-IOV) Support	No	No	Yes	Yes	Yes
The vNIC requirements for later versions of Cisco IOS XE are available in the respective sections in this guide.					

Supported I/O Modes and Drivers

The Cisco CSR 1000v operates within a virtualization environment. Data I/O involves communication between one or more vNICs of the guest OS in which the CSR is operating, and the physical NIC accessed by the host OS.

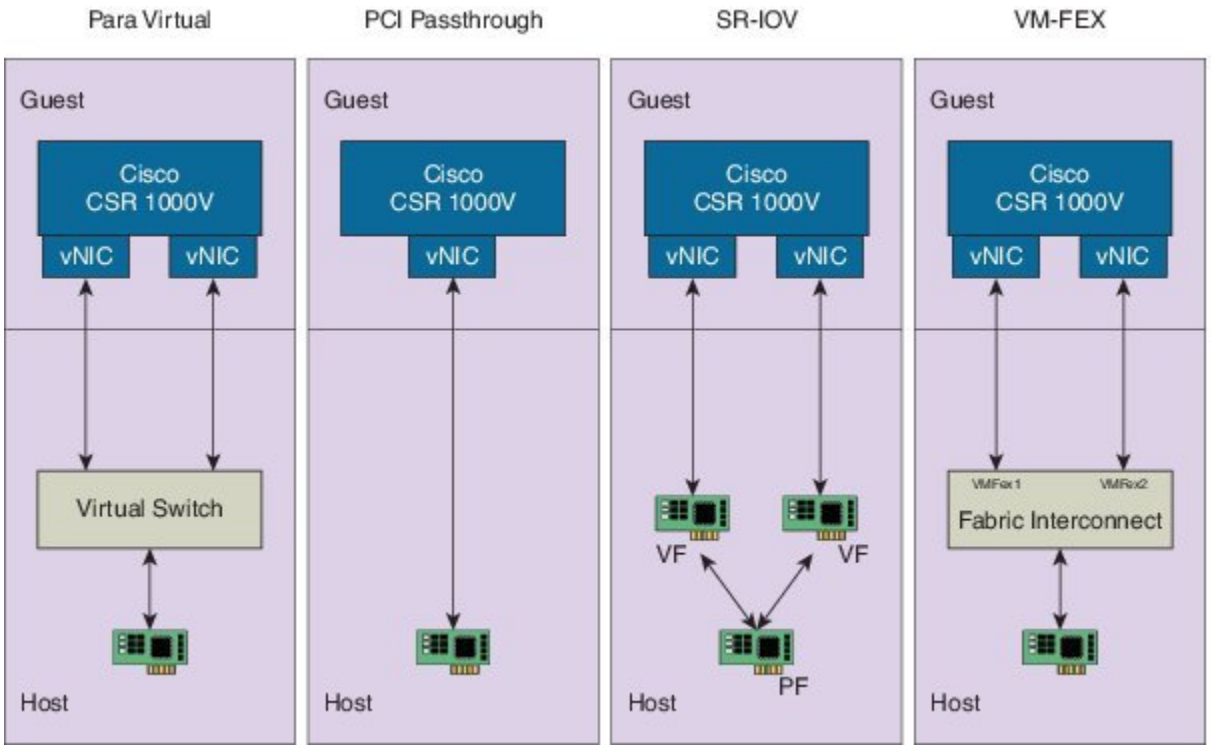
Modes

Beginning with Cisco IOS XE 3.16S and also including Cisco IOS XE Denali 16.3.1 and later, the Cisco CSR 1000v supports several modes of communication between the vNICs and the physical hardware:

- Para Virtual
- PCI Passthrough
- Single Root I/O Virtualization (SR-IOV)
- Cisco Virtual Machine Fabric Extender (VM-FEX)

The figure below, "Cisco CSR 1000v I/O Routing Between vNIC of Guest OS and Hardware NIC of Host", shows the I/O routing options.

Figure 1: Cisco CSR 1000v I/O Routing Between vNIC of Guest OS and Hardware NIC of Host



Drivers

The following table indicates the drivers required to support various I/O modes.

Note See the Hypervisor vNIC requirements to determine the drivers supported for a particular release. PCI Passthrough: enic is not supported in Cisco IOS XE Denali 16.3.1 and higher.

Table 3: Driver Support for I/O Modes

Mode	Cisco CSR1000v Drivers
Para Virtual	• VMXNET3 (ESXi) • Virtio (KVM) • VIF-netfront (Xen) • NetVSC (Hyper-V)
PCI Passthrough	• ixgbe (for Intel 10 gig NIC) • enic (for Cisco VIC)

Mode	Cisco CSR1000v Drivers
SR-IOV	• ixgbevf • i40evf
VM-FEX	Only applicable to Cisco VIC There are 2 modes: • ESXi DirectPath IO: VMXNET3 • PCI Passthrough: enic

Note: For releases Cisco IOS XE 3.16 or later, and Cisco IOS XE Denali 16.3 or later, the boot up process may take a long time (5 minutes) when using passthrough drivers. This is due to performing DHCP during a PXE boot. This issue can be resolved by turning off rom bar for Ethernet PCI devices in the Cisco CSR 1000v xml file; for example:

```
<hostdev mode='subsystem' type='pci' managed='yes'>
     <source>
       <address domain='0x0000' bus='0x85' slot='0x00' function='0x0'/>
     </source>
     <rom bar='off'/>     <------ Add this line to xml file
     <address type='pci' domain='0x0000' bus='0x00' slot='0x07' function='0x0'/>
</hostdev>
```

Limitations

Note See the Hypervisor vNIC requirements to determine the drivers supported for a particular release.

The following table describes the limitations that apply to I/O modes.

Note PCI Passthrough: enic is not supported in Cisco IOS XE Denali 16.3.1 and higher.

Table 4: I/O Mode Limitations

Mode/Driver	Limitations
PCI passthrough (enic)	• Interoperability with another NIC: If enic is connected to other NIC (for example, Intel NIC) and then that NIC is used for other CSR VM (Para virtual or Passthrough), traffic will not pass through if enic is configured with VLAN. • If a VLAN is configured, the other NIC receives a VLAN packet with VLAN id of 0. • Jumbo packet support: In this release, jumbo packet (MTU > 1518) is not supported. • CDP is not supported. • HSRP standby cannot ping the HSRP group address

Mode/Driver	Limitations
SR-IOV (ixgbevf)	• MTU change: (Intel limitation) First change the VF MTU on the host PF using the **ip link set** command. Then change the corresponding interface MTU on the VM. Otherwise, no traffic will pass. (Intel limitation) • MAC address change: After changing the MAC address, it is necessary to change the MAC address of the VF on the host PF using the **ip link set** command. Otherwise, no traffic will pass. (Intel limitation.) • Maximum VLANs: The maximum number of VLANs supported on PF is 64. Together, all VFs can have a total of 64 VLANs. (Intel limitation.) • Maximum Multicast filtering: Intel VF supports registering a maximum of 30 multicast addresses. (Intel limitation.) • Layer2 Learning: The Intel SR-IOV VF does not support promiscuous mode, so Layer 2 functionality, such as EVC, does not work. (Intel limitation.)
SR-IOV (i40evf)	• MTU change: (Intel limitation) First change the VF MTU on the host PF using the **ip link set** command. Then change the corresponding interface MTU on the VM. Otherwise, no traffic will pass. (Intel limitation.) • MAC address change: After changing the MAC address, you must change the MAC address of the VF on the host PF using the **ip link set** command. Otherwise, no traffic will pass. (Intel limitation.) • Maximum VLANs: The maximum number of VLANs supported on PF is 512. Together, all VFs can have a total of 512 VLANs. (Intel limitation.) Per-VF resources are managed by the PF (host) device driver. • Maximum Multicast filtering: The maximum number of mac addresses supported on the PF is 1024. (Intel limitation.) Per-VF resources are managed by the PF (host) device driver. • Layer2 Learning: The Intel SR-IOV VF does not support promiscuous mode, so Layer 2 functionality, such as EVC, does not work. (Intel limitation.) This information about SR-IOV (i40evf) has partly been obtained from Table 7-132. "710 series Versus 82599 Virtualization Support" and Table 7-134. "VF resource allocation" in the Intel Ethernet Controller 710 Series Datasheet.
VM-FEX ESXi DirectPath IO (VMXNET3)	• VLAN is not supported in high-performance mode.

Cisco CSR 1000v and Hypervisor Limitations

This section describes performance limitations due to how the Cisco CSR 1000v integrates with supported hypervisors.

Cisco CSR 1000v and Hypervisor Limitations for Cisco IOS XE Denali 16.3.1 and Later

In these releases, the Cisco CSR 1000v does not support the hot removal of interfaces and does not have the ability to modify vNIC MTU.

Cisco CSR 1000v and Hypervisor Limitations for Cisco IOS XE Denali 16.2

Note Cisco IOS XE Denali 16.3.1 and later is recommended instead of Cisco IOS XE Denali 16.2.

Cisco CSR 1000v and Hypervisor Limitations for Cisco IOS XE Release 3.12S

- When the Cisco CSR 1000v is installed on Microsoft Hyper-V, the interface numbers can change after Microsoft Hyper-V fails over to a new server, or restarts after a live migration.
  - If the server is set to perform ungraceful failover, there is no workaround.
  - If the server is set to perform graceful failover or restart, enter the **clear platform software vnic-if nvtable** command before executing the failover or restart.

This issue is not seen if the maximum number of interfaces is configured.

- When the Cisco CSR 1000v is installed on Microsoft Hyper-V, if you want to configure a VLAN, you must configure the VLAN interfaces on Microsoft Hyper-V using the Hyper-V Power Shell CLI.
- When the Cisco CSR 1000v is installed on Microsoft Hyper-V and an NSF-based virtual hard disk is used, if there is a network connectivity issue between the Cisco CSR 1000v and the NSF server, the Cisco CSR 1000v is unable to use the virtual hard disk even if the network connection is restored. You must reboot the Cisco CSR 1000v to restore access to the virtual hard disk.
- The Microsoft Hyper-V GUI only allows one VLAN to be specified for a Virtual Machine interface. This limits deployments where multiple VLANS for a Virtual Machine interface are used.
- When the MAC address of a Cisco CSR1000v interface is changed from the address assigned by the hypervisor, then traffic to and from external devices is unsuccessful. This occurs even when MAC address spoofing is enabled on the Microsoft Hyper-V vSwitch. Operation of protocols like FHRP, CLNS, and Etherchannel that use their own MAC address may be unsuccessful.
- In Microsoft Hyper-V environments, the following limitations apply when the Windows Power Shell CLI is used to configure VLANs:
  - The power shell CLI commands must be reapplied each time the Cisco CSR1000v is reloaded.
  - When a large AllowedVlanIdList is configured, only lower numbered VLANS may successfully pass traffic. For example, when the following Power Shell CLI command is used:

Set-VMNetworkAdapterVlan -VMName dr-vm-6-1 -Trunk -AllowedVlanIdList 1-2000 -NativeVlanId 0

Only VLANS lower than 300 may successfully pass traffic.

Cisco CSR 1000v and Hypervisor Limitations for Cisco IOS XE Release 3.10S

- Configuring Network Based Application Recognition (NBAR), or Application Visibility and Control (AVC) support on the Cisco CSR 1000v requires a minimum of 4GB of DRAM on the VM, even when using the one vCPU configuration on the VM.
- On the Cisco CSR 1000v, all the NICs are logically named as the Gigabit Ethernet interface. The Cisco CSR 1000v does support the 10G IXGBE vNIC in passthrough mode; but that interface also is also logically named as a Gigabit Ethernet interface. Note that with emulated devices like VMXNET3/PV/VIRTIO from the hypervisor, the Cisco CSR 1000v is not aware of the underlying interfaces. The vSwitch may be connected to a 10-GB physical NIC or 1-GB physical NICs or multiple NICs (with NIC teaming on the hypervisor) as well.

- The Cisco CSR 1000v supports an MTU range from 1500 to 9216 bytes. However, the maximum MTU supported on your hypervisor version may be lower. The MTU value configured on the Cisco CSR 1000v should not exceed the maximum MTU value supported on the hypervisor.

Cisco CSR 1000v and Hypervisor Limitations for Cisco IOS XE Release 3.9S

The following are the Cisco CSR 1000v and VMware ESXi limitations for Cisco IOS XE Release 3.9S:

- The Cisco CSR 1000v interface bandwidth defaults to 1 GB, irrespective of the hypervisor's physical NIC bandwidth. The routing protocols (OSPF, EIGRP) use the Cisco CSR 1000v interface bandwidth values for calculating the costs, not the physical NIC bandwidth.
- When a Cisco CSR 1000v interface is directly connected to a physical router, and that physical router's connecting interface goes down, the change is not reflected on the Cisco CSR 1000v. This is because the Cisco CSR 1000v is actually connected to the hypervisor's vSwitch and the vSwitch uplink port is connected to the physical interface of the router. This behavior is expected.
- The Cisco CSR 1000v provides an MTU range from 1500 to 9216 bytes. However, ESXi 5.0 supports only a maximum value of 9000 bytes.

Server Requirements

The server and processor requirements are different depending on the Cisco CSR 1000v release.

Table 5: Server Requirements

Cisco CSR 1000v Release	Intel	AMD
Cisco IOS XE Release 3.9S	Intel Nehalem and later generation processors	Not supported
Cisco IOS XE Release 3.10S and later	64-bit processors with VT extensions	64-bit processors with VT extensions
Cisco IOS XE Denali 16.3.1 and later	64-bit Intel Core2 and later generation processors with VT extensions and support for Streaming SIMD instructions: SSE, SSE2, SSE3 and SSSE3.	The equivalent of 64-bit Intel Core2 and later generation processors with VT extensions and support for Streaming SIMD instructions: SSE, SSE2, SSE3 and SSSE3.

For more information, see the release notes: http://www.cisco.com/c/en/us/support/routers/cloud-services-router-1000v-series/products-release-notes-list.html.

(For Cisco IOS XE Release 3.9S) the Cisco CSR 1000v uses instructions not supported on Intel pre-Nehalem generation processors. The existence of the required Nehalem or later processor instruction set is determined at boot time. If the required instructions are not present, the following message is displayed:

```
%IOSXEBOOT-4-BOOT_HALT: (rp/0): Halted boot due to missing CPU feature
requirement(s)
```

(For Cisco IOS XE Denali 16.3 and 16.4) the Cisco CSR 1000v uses instructions supported on Intel Core 2 and later generation processors including Streaming SIMD: SSE, SSE2, SSE3 and SSSE3. The existence of the required instruction set is not verified and the deployment of the Cisco CSR 1000v in an environment that does not meet these processor requirements may result in random system reloads.

(For Cisco IOS XE Everest 16.5 and later) the Cisco CSR 1000v uses instructions supported on Intel Core 2 and later generation processors including Streaming SIMD SSE, SSE2, SSE3 and SSSE3. The existence of

the required streaming SIMD instruction sets is determined at boot time. If the required instructions are not present, a message similar to following is displayed:

```
%CPPDRV-3-FATAL_CPU_FEATURE: F0: cpp_driver: CPP0: CPU lacks feature
(Supplemental Streaming SIMD Extensions 3 (SSSE3)). Packet forwarding disabled.
```

Cisco Software Licensing (CSL)

The Cisco CSR 1000v supports two types of license: Cisco Software Licensing and Cisco Smart Licensing. This section summarizes Cisco Software Licensing. For more details of both licensing methods, see Activating Cisco CSR 1000v Licenses, on page 162

The Cisco CSR 1000v supports the following types of Cisco Software License, depending on the software release:

- Perpetual and subscription term licenses for 1, 3, and 5 years based on the following attributes:
  - (Cisco IOS XE 3.13S and later, and Denali 16.3.1 and later) Technology packages: **IPBase**, **Security**, **AX** and **APPX** (supported by Cisco Smart Licensing beginning with Cisco IOS XE 3.15S)
  - Maximum supported throughput level for the **AX** package: 10, 25, 50, 100, 250, or 500 Mbps; 1, 2.5, or 5 Gbps
  - Maximum supported throughput level for the **Security** and **APPX** packages: 10, 25, 50, 100, 250, or 500 Mbps; 1, 2.5, or 5 Gbps
  - Maximum supported throughput level for the **IPBase** package: 10, 25, 50, 100, 250, or 500 Mbps; 1, 2.5, 5, or 10 Gbps
- Memory upgrade licenses (selected technology packages and throughput levels only)
- Evaluation licenses (see Evaluation Licenses for Cisco IOS XE 3.13S and Later and Cisco IOS XE Denali 16.3.1 and Later, on page 38).

Note Three legacy technology packages—**Standard**, **Advanced**, and **Premium**—were replaced in Cisco IOS XE Release 3.13 with the IPBase, Security, and AX technology packages.

The following table lists the available license types for your release.

Table 6: Cisco CSR 1000v Software Licenses

Cisco CSR 1000v Version	License Type	License Term
Cisco IOS XE Release 3.9S	(Legacy) Base subscription technology package licenses (**Standard**, **Advanced**, and **Premium**) for the following throughput maximums: 10 Mbps, 25 Mbps, 50 Mbps	• 1, 3, and 5 years • 60-day evaluation license

Cisco CSR 1000v Version	License Type	License Term
Cisco IOS XE Releases 3.10S, 3.11S	(Legacy) Base subscription **Standard** technology package licenses for the following throughput maximums: 10 Mbps, 50 Mbps, 100 Mbps, 250 Mbps, 500 Mbps, 1 Gbps Base subscription **Advanced** and **Premium** technology package licenses for the following throughput maximums: 10 Mbps, 50 Mbps, 100 Mbps, 250 Mbps Feature Add-on License: • License to add 8 GB of memory with route reflector support. This is available for the Premium or AX packages only. The additional memory is allocated to IOSD processes on the router only. The memory upgrade license does not add available memory on the VM. **Note** Selected licenses are available through a Cisco service representative only.	• 1 and 3 years • Perpetual • 60-day evaluation license
Cisco IOS XE Release 3.12S	(Legacy) Base subscription **Standard** technology package licenses for the following throughput maximums: 10 Mbps, 50 Mbps, 100 Mbps, 250 Mbps, 500 Mbps, 1 Gbps, 2.5 Gbps, 5 Gbps Base subscription **Advanced** and **Premium** technology package licenses for the following throughput maximums: 10 Mbps, 50 Mbps, 100 Mbps, 250 Mbps, 1 Gbps Feature Add-on License: • License to add 8 GB of memory with route reflector support. This is available for the Premium or AX packages only. The additional memory is allocated to IOSD processes on the router only. The memory upgrade license does not add available memory on the VM. Selected licenses are only available through a Cisco service representative.	• 1 and 3 years • Perpetual • 60-day evaluation license

Cisco CSR 1000v Version	License Type	License Term
Cisco IOS XE Release 3.12.1S	New technology package licenses are supported: • **IPBase** package license, with the same feature set as the Standard package • **Security** package license, with the same feature set as the Advanced package • **AX** package license, with the same feature set as the **Premium** package We recommend using these technology packages for compatibility with future releases. All technology packages support the same throughput maximums as feature sets in earlier releases.	• 1 and 3 years • Perpetual • 60-day evaluation license
Cisco IOS XE Releases 3.13S, 3.14S, 3.15S, 3.16S, 3.17 Cisco IOS XE Denali 16.3.1 and later	Base subscription **IPBase** technology package licenses for the following maximum throughputs: 10 Mbps, 50 Mbps, 100 Mbps, 250 Mbps, 500 Mbps, 1 Gbps, 2.5 Gbps, 5 Gbps, 10 Gbps. (IPBase replaces the Standard package.) Base subscription **Security** technology package licenses for the following maximum throughputs: 10, 25, 50, 100, 250, or 500 Mbps; 1, 2.5, or 5 Gbps (**Security** replaces the Advanced package.) Base subscription **AX** technology package licenses for the following maximum throughputs: 10, 25, 50, 100, 250, or 500 Mbps; 1 or 2.5 Gbps (**AX** replaces the Premium package.) Base subscription **Application Experience (APPX)** technology package licenses for the following maximum throughputs: 10 Mbps, 50 Mbps, 100 Mbps, 250 Mbps, 500 Mbps, 1 Gbps, 2.5 Gbps, 5 Gbps	• 1 and 3 years • Perpetual • 60-day evaluation license available through Cisco licensing portal

The supported performance indicates the maximum throughput supported by the Cisco CSR 1000v for the license. If the throughput exceeds the supported performance, the router may experience dropped packets and you will receive notification that the supported performance has been exceeded. The Cisco CSR 1000v uses a performance limiter to regulate the throughput level. For example this applies when using 10 Gbps throughput as part of the IPBase technology package licenses. For more information, see the Configuring an Interface for 10 Gbps Maximum Throughput, on page 167.

If additional performance is required, an additional license for a separate Cisco CSR 1000v VM must be purchased. The Cisco CSR 1000v supports only one router instance per VM.

The Cisco CSR 1000v software licenses operate as follows:

- Each software license can be used for only one VM.
- You can install more than one license on a VM, but the multiple licenses can only apply to that VM.

- Similar to Cisco hardware products, the software license is node-locked to the unique device identifier (UDI) of that product. The Cisco CSR 1000v generates a Virtual UDI (vUDI) when first installed on the VM, and licenses are node-locked to that vUDI. One license per VM instance is required. Instances that are cloned from a repository must generate a new vUDI.

Note When you clone the Cisco CSR 1000v, you will automatically get a new vUDI, and all the licenses from the original VM should be removed.

- You must purchase and install a new technology package license if you want to upgrade or downgrade the technology level. For example, if you have a Premium technology package license and you want to downgrade to the Standard technology package, you must purchase a new Standard technology package license.
- In Cisco IOS XE Release 3.10S, the default license will not enable advanced IPsec features and MPLS.
- The Cisco CSR 1000v does not provide or support Right-to-Use performance licenses.
- You will receive warning notices that the subscription term license will expire beginning eight weeks before license expiration.

The licenses must be activated in order for the Cisco CSR 1000v network ports to provide the supported throughput.

When the Cisco CSR 1000v is first booted, the router operates in evaluation mode, and provides limited feature support and limited throughput. To obtain the full feature support and throughput provided by your license, you must install the license using the **license install** command. The configuration requirements depend on the release version:

- In Cisco IOS XE 3.12S and earlier, to access the features supported in your license, you must enter the **license boot level** command and set it to the level supported by your license. The Cisco CSR 1000v must be rebooted for the new license level to take effect and to have the new license applied.
- In Cisco IOS XE 3.13S and later, the Cisco CSR 1000v first boots up in the AX technology mode by default, so all features in this package are supported. Installing an AX technology license applies the AX license immediately, and the throughput is increased to the maximum throughput of the installed license. Rebooting the router is not required.

If you install a different technology license (IPBase, Security or APPX), the corresponding license boot level command setting is automatically added to the running configuration, but you must reboot the router for the new license technology level to take effect and to have the license applied.

The installed license technology package must match the router's current technology level (as shown with the **show version** command). If the license package does not match the current license level the throughput is limited to 100kbps. To apply a license belonging to a different technology package level, you must update the license level using the license boot level command and reboot the Cisco CSR 1000v for the new license level to take effect.

If the throughput license expires or becomes invalid, the maximum throughput of the router reverts to 2.5 Mbps (Cisco IOS XE 3.12S and earlier), or 100 Kbps (Cisco IOS XE 3.13S and later), upon reload.

The subscription term begins on the day the license is issued.

For more information about license activation, see the Activating Cisco CSR 1000v Licenses, on page 162.

If you rehost the Cisco CSR 1000v to a VM on another server, the following rules apply:

- You must purchase a new rehost software license that lasts for the period remaining on the original license.
- If the original license was renewed, the rehosted software license will last for the period remaining on the renewed license.
- You have a 60-day grace period to remove the software license from the original server hardware and activate it on the rehosted server hardware.

The Cisco CSR 1000v also supports Cisco License Manager and Cisco License Call Home. For more information about the standard Cisco IOS XE software activation procedure, and information about Cisco License Manager and Cisco License Call Home, see the Software Activation Configuration Guide, Cisco IOS XE Release 3S .

Evaluation Licenses for Cisco IOS XE 3.13S and Later and Cisco IOS XE Denali 16.3.1 and Later

Evaluation licenses are available to try out Cisco CSR 1000v features. Evaluation licenses are obtained differently depending on the Cisco IOS XE release version. This section describes versions Cisco IOS XE 3.13S or later and Cisco IOS XE Denali 16.3.1 or later.

Default

Beginning with the Cisco IOS XE 3.13S release, the CSR 1000v boots by default with the following features:

- AX technology package features
- 100 Kbps maximum throughput

Evaluation License Options

Evaluation licenses valid for 60 days are available at the Cisco licensing portal.

http:/www.cisco.com/go/license

The evaluation license options enable test driving additional technology packages and higher throughputs. (The throughputs available through evaluation licenses are the highest supported throughput levels for the package type.)

- IPBase Technology package, 10 Gbps
- SEC Technology package, 5 Gbps
- APP Technology package, 5 Gbps
- AX Technology package, 2.5 Gbps
- 1000 broadband sessions
- 12 GB memory upgrade

Testing a Lower Maximum Throughput

To test a lower throughput license type not listed here, use the **platform hardware throughput level MB <throughput>** command to set the throughput to a supported level below that provided by the installed license.

This has the same effect as installing a license for that throughput level. For example, on a CSR 1000v with a 5 Gbps license installed, the following command sets the throughput level to 250 Mbps:

```
platform hardware throughput level MB 250
```

The supported throughput levels are: 10 Mbps, 50 Mbps, 100 Mbps, 250 Mbps, 500 Mbps, 1 Gbps, 2.5 Gbps, 5 Gbps, 10 Gbps

For any additional questions, contact your Cisco sales representative.

Obtaining an Evaluation License from the Cisco Licensing Portal

To obtain a 60-day evaluation license for the Cisco CSR 1000v, follow the instructions below.

When the 60-day evaluation license expires, the maximum throughput becomes limited to 100 Kbps upon reload. For more information, see Installing CSL Evaluation Licenses for Cisco IOS XE 3.13S and Later, on page 162.

Note These instructions are subject to change.

Before you begin

Procedure

Step 1 Navigate to https://www.cisco.com/go/license and log in.

Step 2 Navigate to the Product License Registration Portal.

Step 3 On the Product License Registration page, select “Continue to Product License Registration.”

Step 4 Click “Get Other Licenses” and select “Demo and Evaluation” from the dropdown menu.

Step 5 In the Product Family section, select “Routers & Switches.” In the Product section, select “Cisco Cloud Services Router 1000v.” Click Next.

Step 6 Select the desired license type. Enter the UDI Serial number, then click Next to generate the license. You can display the UDI Serial number on your router by entering the **show license udi** command.

Evaluation Licenses for Cisco IOS XE 3.12S and Earlier

Evaluation licenses are available to try out Cisco CSR 1000v features. Evaluation licenses are obtained differently depending on the IOS XE release version. This section describes versions Cisco IOS XE 3.12S or earlier.

Prior to the Cisco IOS XE 3.13S release, the Cisco CSR 1000v came bundled with a 60-day evaluation license included with the software image, providing:

- Premium technology package features
- 50 Mbps maximum throughput

The license is activated by entering the **license boot level** command and rebooting the router.

When the 60-day evaluation license expires, the maximum throughput reverts to 2.5 Mbps and to the Standard feature set upon reload.

Cisco Smart Licensing

The Cisco CSR 1000v supports two types of license: Cisco Software Licensing and Cisco Smart Licensing. This section summarizes Cisco Smart Licensing. For details, see Cisco Smart Licensing, on page 179

Beginning with Cisco IOS XE Release 3.15S, the Cisco CSR 1000v supports activation using Cisco Smart Licensing (CSL). To use Cisco Smart Licensing, first configure the Call Home feature and obtain Cisco Smart Call Home Services. For details, see Cisco Smart Licensing, on page 179.

The Cisco CSR 1000v supports the following license types (Cisco IOS XE 3.14S and later):

- IPBase
- Security
- AX
- APPX

Differences Between Cisco CSR 1000v Series and ASR 1000 Series

Unlike traditional Cisco hardware router platforms, the Cisco CSR 1000V Series is a virtual router that runs independently on an x86 machine. As a result, the Cisco CSR 1000v Series architecture has unique attributes that differentiate it from hardware-based router platforms.

For example, the table below lists a comparison of some key areas where the Cisco CSR 1000v Series differs from the Cisco ASR 1000 series routers.

Table 7: Cisco CSR 1000v Series Architecture Differences with Cisco ASR 1000 Series Routers

Feature	Cisco ASR 1000 Series	Cisco CSR 1000v Series
Hard Disk	Supported.	The Cisco CSR 1000v does not include a hard disk. The software image is stored on bootflash only (8 GB).
Physical resources	Managed by architecture of the hardware platform.	Managed by the hypervisor. Physical resources are shared among VMs.
Console types supported	Physical serial port.	• Virtual VGA console • Virtual serial port network option (virtual terminal server) • Named pipe option • Physical serial port on the ESXi or KVM host

Feature	Cisco ASR 1000 Series	Cisco CSR 1000v Series
ROMMON	Supported.	The Cisco CSR 1000v does not include ROMMON, but uses GRUB to provide similar but more limited functionality.
Break Signal	Supported.	Not supported.
Port numbering	See the Cisco ASR1000 documentation .	Gigabit Ethernet x ports only.
ISSU	Supports In-Service Software Upgrades (ISSU).	Not supported.
Subpackage upgrades	Supports installation of subpackages for specific SPAs and SIP SPAs.	Subpackages not supported. The Cisco CSR 1000v does not support SPAs.
Diagnostic mode	Supported.	Not supported.
Dynamic addition/deletion of ports	Supported.	Supported. (Requires reload of the VM.)

Supported Cisco IOS XE Technologies

The Cisco CSR 1000v Series Cloud Services Router supports selected Cisco IOS XE technologies. The Cisco CSR 1000v supports a more limited set of functionality compared to other router platforms.

The table below lists the major Cisco IOS XE technologies the Cisco CSR 1000V supports. Technologies not listed are not currently supported on the Cisco CSR 1000v. Not all features in a given technology may be supported. To verify support for specific features, use the Cisco Feature Navigator.

The information listed in this table applies only if using the Cisco IOS XE CLI. Support for Cisco IOS XE technologies is more limited in the following scenarios:

- When you deploy Cisco CSR 1000v on Amazon Web Services (AWS). For more information, see Cisco CSR 1000V Series Cloud Services Router Deployment Guide for Amazon Web Services.

  See also the following white paper that includes information about the high availability solution: Deploying the Cisco Cloud Services Router 1000V Series in Amazon AWS: Design and Implementation Guide.

- When you deploy Cisco CSR 1000v on Microsoft Azure. For more information, see Cisco CSR 1000v Deployment Guide for Microsoft Azure.

- When using the Cisco IOS XE REST API to manage the Cisco CSR 1000v. For more information, see Enabling Management by REST API, on page 317.

  For information about Cisco IOS XE technologies supported by the REST API, see the Cisco IOS XE REST API Management Reference Guide .

- When using Cisco Prime Network Services Controller (PNSC) to remotely manage the Cisco CSR 1000v. For more information on features supported, see Configuring Support for Remote Management by the Cisco Prime Network Services Controller, on page 343.

Note The IPBase, Security, and AX license technology packages became available beginning with Cisco IOS XE 3.12.1.

Table 8: Cisco IOS XE Technologies Supported on the Cisco CSR 1000v Cloud Services Router

Technologies Supported	Technology Package Licenses Supported in Cisco IOS XE Releases 3.12S and Earlier (Legacy)	Technology Package Licenses Supported in Cisco IOS XE Releases 3.13S and Later, and Denali 16.3.1 and Later	See the Following Documentation:
IP:			
• IPv4 Routing • IPv4 Fragmentation and Reassembly • IPv6 Forwarding	• Standard • Advanced • Premium	• IPBase • Security • AX • APPX	• IP Addressing Configuration Guide Library, Cisco IOS XE Release 3S • Cisco IOS IP Addressing Services Command Reference
• IPv6 Routing	• Standard • Advanced • Premium	• IPBase • Security • AX • APPX	• IPv6 Configuration Guide Library, Cisco IOS XE Release 3S • Cisco IOS IPv6 Command Reference
• Generic Routing Encapsulation (GRE)	• Standard • Advanced • Premium	• IPBase • Security • AX • APPX	• Interface and Hardware Component Configuration Guide, Cisco IOS XE Release 3S • Cisco IOS Interface and Hardware Component Command Reference
• LISP	• Premium	• AX • APPX	• IP Routing: LISP Configuration Guide, Cisco IOS XE Release 3S • Cisco IOS IP Routing: LISP Command Reference

Technologies Supported	Technology Package Licenses Supported in Cisco IOS XE Releases 3.12S and Earlier (Legacy)	Technology Package Licenses Supported in Cisco IOS XE Releases 3.13S and Later, and Denali 16.3.1 and Later	See the Following Documentation:
• Connectionless mode network service (CLNS)	—	• IPBase • Security • AX • APPX	• ISO CLNS Configuration Guide, Cisco IOS XE Release 3S
Basic Routing:			
• BGP	• Standard • Advanced • Premium	• IPBase • Security • AX • APPX	• IP Routing: BGP Configuration Guide, Cisco IOS XE Release 3S • Cisco IOS IP Routing: BGP Command Reference
• EIGRP	• Standard • Advanced • Premium	• IPBase • Security • AX • APPX	• IP Routing: EIGRP Configuration Guide, Cisco IOS XE Release 3S • Cisco IOS IP Routing: EIGRP Command Reference
• ISIS	• Standard • Advanced • Premium	• IPBase • Security • AX • APPX	• IP Routing: ISIS Configuration Guide, Cisco IOS XE Release 3S • Cisco IOS IP Routing: ISIS Command Reference
• OSPF	• Standard • Advanced • Premium	• IPBase • Security • AX • APPX	• IP Routing: OSPF Configuration Guide, Cisco IOS XE Release 3S • Cisco IOS IP Routing: OSPF Command Reference
• Performance Routing	• Standard • Advanced • Premium	• IPBase • Security • AX • APPX	• Performance Routing Configuration Guide, Cisco IOS XE Release 3S • Cisco IOS Performance Routing Command Reference

Technologies Supported	Technology Package Licenses Supported in Cisco IOS XE Releases 3.12S and Earlier (Legacy)	Technology Package Licenses Supported in Cisco IOS XE Releases 3.13S and Later, and Denali 16.3.1 and Later	See the Following Documentation:
IP Multicast:			
• IGMP	• Advanced • Premium	• Security • AX	• IP Multicast: IGMP Configuration Guide, Cisco IOS XE Release 3S • Cisco IOS IP Multicast Command Reference
• PIM	• Advanced • Premium	• Security • AX	• IP Multicast: PIM Configuration Guide, Cisco IOS XE Release 3S • Cisco IOS IP Multicast Command Reference
IP Switching:			
• Cisco Express Forwarding	• Standard • Advanced • Premium	• IPBase • Security • AX • APPX	• IP Switching Cisco Express Forwarding Configuration Guide, Cisco IOS XE Release 3S • Cisco IOS IP Switching Command Reference
Wide Area Networking:			
• OTV (Supported beginning in Cisco IOS XE 3.10S.)	• Premium	• AX • APPX	• Wide-Area Networking Configuration Guide: Overlay Transport Virtualization, Cisco IOS XE Release 3S • Cisco IOS Wide-Area Networking Command Reference
• VxLAN (Supported beginning in Cisco IOS XE 3.11S.)	• Premium	• AX • APPX	• Cisco CSR 1000V VxLAN Support
• WCCPv2	• Premium	• AX • APPX	• IP Application Services Configuration Guide, Cisco IOS XE Release 3S • Cisco IOS IP Application Services Command Reference

Technologies Supported	Technology Package Licenses Supported in Cisco IOS XE Releases 3.12S and Earlier (Legacy)	Technology Package Licenses Supported in Cisco IOS XE Releases 3.13S and Later, and Denali 16.3.1 and Later	See the Following Documentation:
VPN:			
• IPsec VPN	• Advanced • Premium	• Security • AX	• Secure Connectivity Configuration Guide Library, Cisco IOS XE Release 3S
• DMVPN	• Advanced • Premium	• Security • AX	• Dynamic Multipoint VPN Configuration Guide, Cisco IOS XE Release 3S
• Easy VPN	• Advanced • Premium	• Security • AX	• Easy VPN Configuration Guide, Cisco IOS XE Release 3S
• FlexVPN	• Advanced • Premium	• Security • AX	• FlexVPN and Internet Key Exchange Version 2 Configuration Guide, Cisco IOS XE Release 3S
• GETVPN (Supported beginning in Cisco IOS XE Everest 16.6.1)	• Advanced • Premium	• Security • AX	• Cisco Group Encrypted Transport VPN Configuration Guide, Cisco IOS XE Release 3S
• SSL VPN (Supported beginning in Cisco IOS XE 3.12.1S.)	• Advanced • Premium	• Security • AX	• SSL VPN Configuration Guide, Cisco IOS XE Release 3S
MPLS:			
• MPLS	• Premium	• APPX • AX	See the Multiprotocol Label Switching (MPLS) guides in the CSR 1000v Configuration Guides.
• EoMPLS	• Premium	• APPX • AX	• See the Multiprotocol Label Switching (MPLS) guides in the CSR 1000v Configuration Guides.

Technologies Supported	Technology Package Licenses Supported in Cisco IOS XE Releases 3.12S and Earlier (Legacy)	Technology Package Licenses Supported in Cisco IOS XE Releases 3.13S and Later, and Denali 16.3.1 and Later	See the Following Documentation:
• VRF	• Advanced • Premium	• IPBase	• MPLS: Layer 3 VPNs Configuration Guide, Cisco IOS XE Release 3S • Cisco IOS Multiprotocol Label Switching Command Reference
• VPLS (Supported beginning in Cisco IOS XE 3.10S.)	• Premium	• APPX • AX	• MPLS Layer 2 VPNs Configuration Guide, Cisco IOS XE Release 3S • Cisco IOS Multiprotocol Label Switching Command Reference
Network Management:			
• SNMP	• Standard • Advanced • Premium	• IPBase • Security • AX • APPX	• SNMP Configuration Guide, Cisco IOS XE Release 3S • Cisco IOS Network Management Command Reference
• Flexible NetFlow	• Standard • Advanced • Premium	• IPBase • Security • AX • APPX	• Flexible NetFlow Configuration Guide, Cisco IOS XE Release 3S • Cisco IOS Flexible NetFlow Command Reference
• Secure Shell (SSH)	• Standard • Advanced • Premium	• IPBase • Security • AX • APPX	• Secure Shell Configuration Guide, Cisco IOS XE Release 3S
QoS:			

Technologies Supported	Technology Package Licenses Supported in Cisco IOS XE Releases 3.12S and Earlier (Legacy)	Technology Package Licenses Supported in Cisco IOS XE Releases 3.13S and Later, and Denali 16.3.1 and Later	See the Following Documentation:
• QoS	• Standard (Cisco IOS XE 3.12S) • Advanced(Cisco IOS XE 3.10S and later) • Premium	• IPBase • Security • AX • APPX	• Quality of Service Solutions Configuration Guide Library, Cisco IOS XE Release 3S • Cisco IOS Quality of Service Solutions Command Reference
Services:			
• NAT	• Standard • Advanced • Premium	• IPBase • Security • AX • APPX	• IP Addressing: NAT Configuration Guide, Cisco IOS XE Release 3S • Cisco IOS IP Addressing Services Command Reference
Access Control:			
• AAA	• Standard • Advanced • Premium	• IPBase • Security • AX • APPX	• Authentication Authorization and Accounting Configuration Guide, Cisco IOS XE Release 3S
• Access Control Lists	• Standard • Advanced • Premium	• IPBase • Security • AX • APPX	• Securing the Data Plane Configuration Guide Library, Cisco IOS XE Release 3S
• IP SLA	• Premium	• AX • APPX	• IP SLAs Configuration Guide, Cisco IOS XE Release 3S • Cisco IOS IP SLAs Command Reference

Technologies Supported	Technology Package Licenses Supported in Cisco IOS XE Releases 3.12S and Earlier (Legacy)	Technology Package Licenses Supported in Cisco IOS XE Releases 3.13S and Later, and Denali 16.3.1 and Later	See the Following Documentation:
• RADIUS	• Standard • Advanced • Premium	• IPBase • Security • AX • APPX	• RADIUS Configuration Guide Cisco IOS XE Release 3S
• TACACS+	• Standard • Advanced • Premium	• IPBase • Security • AX • APPX	• TACACS+ Configuration Guide Cisco IOS XE Release 3S
• Layer3 Firewall	• Advanced • Premium	• Security • AX	• MPLS: Layer 3 VPNs Configuration Guide, Cisco IOS XE Release 3S • Cisco IOS Multiprotocol Label Switching Command Reference
• Zone-Based Firewall	• Advanced • Premium	• Security • AX	• Security Configuration Guide: Zone-Based Policy Firewall, Cisco IOS XE Release 3S
• Zone-Based Firewall Multi-tenancy for Cloud Integrated Security Solution (Supported starting with Cisco IOS XE Denali 16.4.1.)	• NA	• Advanced • Premium	• Cloud Integrated Security Solution Guide
Application Services:			
• Application Visibility and Control (AVC)	• Premium	• AX • APPX	• Application Visibility and Control Configuration Guide

Technologies Supported	Technology Package Licenses Supported in Cisco IOS XE Releases 3.12S and Earlier (Legacy)	Technology Package Licenses Supported in Cisco IOS XE Releases 3.13S and Later, and Denali 16.3.1 and Later	See the Following Documentation:
• NBAR2	• Premium	• AX • APPX	• NBAR Protocol Library, Cisco IOS XE Release 3S • QoS: NBAR Configuration Guide, Cisco IOS XE Release 3S Download the NBAR2 protocol pack for your release on the Cisco CSR 1000V software download page. For more information, see the NBAR2 Protocol Library.
Broadband:			
• Broadband Network Gateway (Supported beginning in Cisco IOS XE 3.13S.)	NA	• APPX (Requires broadband add-on feature license (L-CSR-BB-1K=).	• Broadband Access Aggregation and DSL Configuration Guide, Cisco IOS XE Release 3S • Cisco IOS Broadband Access Aggregation and DSL Command Reference
• Intelligent Services Gateway (Supported beginning in Cisco IOS XE 3.13S.)	NA	• APPX (Requires broadband add-on feature license (L-CSR-BB-1K=).	• Intelligent Services Gateway Configuration Guide, Cisco IOS XE Release 3S • Cisco IOS Intelligent Services Gateway Command Reference
Redundancy:			
• HSRP	• Standard • Advanced • Premium	• IPBase • Security • AX • APPX	• First Hop Redundancy Protocols Configuration Guide, Cisco IOS XE Release 3S
WAAS:			
• Integrated AppNav-XE	• Premium	• AX • APPX	• Configuration Guide for AppNav-XE for Cisco Cloud Services Router 1000V Series

Management Support

Managing the Router Using Cisco Configuration Professional

Beginning with Cisco IOS XE Release 3.12S, the Cisco CSR 1000v supports managing the router using Cisco Configuration Professional. The minimum version required is Cisco Configuration Professional 2.8. For more information, see the Cisco Configuration Professional documentation.

Managing the Router Using the Cisco IOS XE REST API

Beginning with Cisco IOS XE Release 3.10S, and including Cisco IOS XE Denali 16.2, a REST API is available as an alternative method for managing the Cisco CSR 1000v router.

Note The Cisco CSR 1000v currently does not fully support IPv6 for the REST API.

The following requirements apply to the Cisco IOS XE REST API (formerly called the Cisco CSR 1000v REST API):

- The Cisco IOS XE REST API supports only selected features and technologies compared to the Cisco IOS XE command-line interface.
- The REST API is supported over HTTPS only.
- (Cisco IOS XE releases 3.13.2, 3.14.1, 3.15 and later, and Denali 16.3.1 and later) REST API (and PNSC) support is limited to TLS.
- The Cisco CSR 1000v Amazon Machine Image (AMI) does not support management of the router using the REST API.

For more information about configuring the router to support management using the REST API, see Enabling Management by REST API, on page 317. For more information about using the Cisco IOS XE REST API, see the Cisco IOS XE REST API Management Reference Guide .

Managing the Router Using Cisco Prime Network Services Controller

Beginning with Cisco IOS XE Release 3.11S, you can use the Cisco Prime Network Services Controller to provision, manage, and monitor the Cisco CSR 1000v. Cisco Prime Network Services Controller can be used to streamline configuration when you are provisioning and managing many Cisco CSR 1000v VMs.

If deploying the Cisco CSR 1000v on ESXi, support for remote management using PNSC can be configured while deploying the OVA template. If deploying the Cisco CSR 1000v on other hypervisors, or if launching the Cisco CSR 1000v on an AWS instance, the PNSC configuration settings are performed using the Cisco IOS CLI.

For more information about remote management using Cisco Prime Network Services Controller, see:

Configuring the Management Interface to Support Remote Management by the Cisco Prime Network Services Controller, on page 343

Enabling Remote Management by the Cisco Prime Network Services Controller Host, on page 346

Disabling Remote Management by the Cisco Prime Network Services Controller Host, on page 348

For more information about configuring Cisco Prime Network Services Controller and using the GUI for remote management, see the following documentation:

- Cisco Prime Network Services Controller Quick Start Guide
- Cisco Prime Network Services Controller User Guide

The table below lists the Cisco Prime Network Services Controller versions that are compatible with the Cisco CSR 1000v.

Table 9: Cisco CSR 1000v Compatibility with Cisco Prime Network Services Controller

Cisco IOS XE Release for Cisco CSR 1000V	Cisco Prime Network Services Controller Version	Hypervisors Supported for Implementation	Feature Support
Cisco IOS XE Release 3.11S	Version 3.2.1Version 3.2.2	• VMware ESXi • KVM	Baseline features: • Hostname, DNS, User Credentials • Interfaces: cloud-facing, external-facing • Interface types: Gigabit Ethernet, loopback • NAT, NTP • ACL, Firewall • Routing: BGP, OSPF, static routes • Syslog
Cisco IOS XE Release 3.12S and later	Version 3.2.1 Version 3.2.2	• VMware ESXi • KVM	Features added in this release: • Sub-interface • IPSec VPN • DHCP Server/Relay • Routing: EIGRP • SNMP • NAT: Overload, PAT • VPN Tunnel interface • Site-to-Site VPN

Cisco Unified Computing System (UCS) Products

Table 10: Cisco CSR 1000v Compatibility with Cisco UCS Servers

Cisco IOS XE Release 3.9S and later:	
Cisco Unified Computing System (UCS) Products	The Cisco UCS server requirements are: • VMware-certified • 4 or more cores configured • 6 GB or more memory • VMware vCenter or standalone VMware vSphere client installed to manage the ESXi server See http://www.cisco.com/en/US/products/ps10477/prod_technical_reference_list.html to determine the UCS hardware and software that is compatible with the supported hypervisors.

Finding Support Information for Platforms and Cisco Software Images

Cisco software is packaged in feature sets consisting of software images that support specific platforms. The feature sets available for a specific platform depend on which Cisco software images are included in a release. To identify the set of software images available in a specific release or to find out if a feature is available in a given Cisco IOS XE software image, you can use Cisco Feature Navigator, the Software Advisor, or the Cisco CSR 1000v Release Notes.

Using Cisco Feature Navigator

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS XE software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn . An account on Cisco.com is not required.

Using the Software Advisor

To see if a feature is supported by a Cisco IOS XE release, to locate the software document for that feature, or to check the minimum Cisco IOS XE software requirements with your router, Cisco maintains the Software Advisor tool on Cisco.com at:

http://tools.cisco.com/Support/Fusion/FusionHome.do

You must be a registered user on Cisco.com to access this tool.

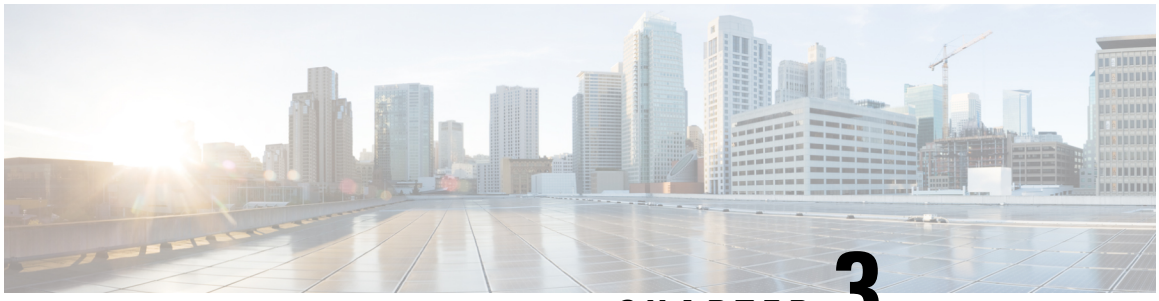
Using the Software Release Notes

Cisco IOS XE software release notes provide the following information:

- Platform support
- Memory recommendations
- New features
- Open and resolved severity 1 and 2 caveats

Release notes are intended to be release-specific for the most current release, and the information provided in these documents may not be cumulative in providing information about features that first appeared in previous releases. See Cisco Feature Navigator for cumulative feature information.

For more information, see http://www.cisco.com/c/en/us/support/routers/cloud-services-router-1000v-series/products-release-notes-list.html.



CHAPTER 3

Using Cisco IOS XE Software

Using Cisco IOS XE Software

This chapter provides information about the Cisco IOS XE software used to configure the Cisco CSR 1000v and Cisco Cisco ISRv. The software for the Cisco CSR 1000v and Cisco ISRv uses standard Cisco IOS XE CLI commands and conventions.

Commands are not case sensitive. You can abbreviate commands and parameters if the abbreviations contain enough letters to be different from any other currently available commands or parameters.

The table below lists the keyboard shortcuts for entering and editing commands.

Table 11: Keyboard Shortcuts

Keystrokes	Purpose
Ctrl-B or the **Left Arrow** key	Move the cursor back one character.
Ctrl-F orthe **Right Arrow** key	Move the cursor forward one character.
Ctrl-A	Move the cursor to the beginning of the command line.
Ctrl-E	Move the cursor to the end of the command line.
Esc B	Move the cursor back one word.
Esc F	Move the cursor forward one word.

The history buffer stores the last 10 commands you entered. History substitution allows you to access these commands without retyping them, by using special abbreviated commands.

Table 12: History Substitution Commands

Command	Purpose
Ctrl-P or the **Up Arrow** key	Recall commands in the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Ctrl-N or the **Down Arrow** key	Return to more recent commands in the history buffer after recalling commands with **Ctrl-P** or the **Up Arrow** key.
Router# **show history**	While in EXEC mode, list the last several commands you have just entered.

The command modes available in the traditional Cisco IOS CLI are exactly the same as the command modes available in Cisco IOS XE.

Use the CLI to access Cisco IOS XE software. Because the CLI is divided into many different modes, the commands available to you at any given time depend on the mode that you are currently in. Entering a question mark (**?**) at the CLI prompt allows you to obtain a list of commands available for each command mode.

When you log in to the CLI, you are in user EXEC mode. User EXEC mode contains only a limited subset of commands. To have access to all commands, enter privileged EXEC mode, normally by using a password. From privileged EXEC mode, you can issue any EXEC command—user or privileged mode—or you can enter global configuration mode. Most EXEC commands are one-time commands. For example, **show** commands show important status information, and **clear** commands clear counters or interfaces. The EXEC commands are not saved when the software reboots.

Configuration modes allow you to make changes to the running configuration. If you later save the running configuration to the startup configuration, these changed commands are stored when the software is rebooted. To enter specific configuration modes, you must start at global configuration mode. From global configuration mode, you can enter interface configuration mode and a variety of other modes, such as protocol-specific modes.

The table below describes how to access and exit various common command modes of the Cisco IOS XE software. It also shows examples of the prompts displayed for each mode.

Table 13: Accessing and Exiting Command Modes

Command Mode	Access Method	Prompt	Exit Method
User EXEC	Log in.	`Router>`	Use the **logout** command.
Privileged EXEC	From user EXEC mode, use the **enable** EXEC command.	`Router#`	To return to user EXEC mode, use the **disable** command.
Global configuration	From privileged EXEC mode, use the **configure terminal** privileged EXEC command.	`Router(config)#`	To return to privileged EXEC mode from global configuration mode, use the **exit** or **end** command.

Command Mode	Access Method	Prompt	Exit Method
Interface configuration	From global configuration mode, specify an interface using an **interface** command.	`Router(config-if)#`	To return to global configuration mode, use the **exit** command. To return to privileged EXEC mode, use the **end** command.

Entering a question mark (**?**) at the CLI prompt displays a list of commands available for each command mode. You can also get a list of keywords and arguments associated with any command by using the context-sensitive help feature.

To get help specific to a command mode, a command, a keyword, or an argument, use one of the commands listed in the table below.

Table 14: Help Commands and Purpose

Command	Purpose
help	Provides a brief description of the help system in any command mode.
abbreviated-command-entry?	Provides a list of commands that begin with a particular character string. (No space between command and question mark.)
abbreviated-command-entry<Tab>	Completes a partial command name.
?	Lists all commands available for a particular command mode.
command ?	Lists the keywords or arguments that you must enter next on the command line. (Space between command and question mark.)

Finding Command Options

This section provides an example of how to display syntax for a command. The syntax can consist of optional or required keywords and arguments. To display keywords and arguments for a command, enter a question mark (**?**) at the configuration prompt or after entering part of a command followed by a space. The Cisco IOS XE software displays a list and brief description of available keywords and arguments. For example, if you were in global configuration mode and wanted to see all the keywords or arguments for the **arap** command, you would type **arap ?**.

The <cr> symbol in command help output stands for "carriage return." On older keyboards, the carriage return key is the Return key. On most modern keyboards, the carriage return key is the Enter key. The <cr> symbol at the end of command help output indicates that you have the option to press **Enter** to complete the command and that the arguments and keywords in the list preceding the <cr> symbol are optional. The <cr> symbol by itself indicates that no more arguments or keywords are available and that you must press **Enter** to complete the command.

The following examples show how you can use the question mark (**?**) to assist you in entering commands.

Table 15: Finding Command Options

Command	Comment

`Router> enable`
`Password: <password>`
`Router#`
	Enter the **enable** command and password to access privileged EXEC commands. You are in privileged EXEC mode when the prompt changes to a "# " from the "> "; for example, Router> to Router# .

`Router#`
`configure terminal`
`Enter configuration commands, one per line. End with CNTL/Z.`
`Router(config)#`
	Enter the **configure terminal** privileged EXEC command to enter global configuration mode. You are in global configuration mode when the prompt changes to Router(config)# .

`Router(config)# interface GigabitEthernet ?`
`  <0-6>  GigabitEthernet interface number`
`Router(config)# interface GigabitEthernet 1`
`Router(config-if)#`
	Enter interface configuration mode by specifying the serial Gigabit Ethernet interface that you want to configure using the **interface GigabitEthernet** *number* global configuration command.
	Enter **?** to display what you must enter next on the command line.
	When the <cr> symbol is displayed, you can press Enter to complete the command.
	You are in interface configuration mode when the prompt changes to Router(config-if)# .
	Note
	The Cisco CSR 1000v and Cisco ISRv support only Gigabit Ethernet interfaces.

Command	Comment

`Router(config-if)# ?`
`Interface configuration commands:`
`  .`
`  .`
`  .`
`  ip  Interface Internet Protocol config commands`
`  keepalive  Enable keepalive`
`  lan-name  LAN Name command`
`  llc2  LLC2 Interface Subcommands`
`  load-interval  Specify interval for load calculation for an interface`
`  locaddr-priority  Assign a priority group`
`  logging  Configure logging for interface`
`  loopback  Configure internal loopback on an interface`
`  mac-address  Manually set interface MAC address`
`  mls  mls router sub/interface commands`
`  mpoa  MPOA interface configuration commands`
`  mtu  Set the interface Maximum Transmission Unit (MTU)`
`  netbios  Use a defined NETBIOS access list or enable name-caching`
`  no  Negate a command or set its defaults`
`  nrzi-encoding  Enable use of NRZI encoding`
`  ntp  Configure NTP`
`  .`
`  .`
`  .`
`Router(config-if)#`
	Enter **?** to display a list of all the interface configuration commands available for the Gigabit Ethernet interface. This example shows only some of the available interface configuration commands.

Command	Comment

`Router(config-if)# ip ?`
`Interface IP configuration subcommands:`
`  access-group  Specify access control for packets`
`  accounting  Enable IP accounting on this interface`
`  address  Set the IP address of an interface`
`  authentication  authentication subcommands`
`  bandwidth-percent  Set EIGRP bandwidth limit`
`  bgp  BGP interface commands..<snipped for brevity>`
`  .`
`  .`
`  .`
`Router(config-if)# ip`
	Enter the command that you want to configure for the interface. This example uses the **ip** command.
	Enter **?** to display what you must enter next on the command line. This example shows only some of the available interface IP configuration commands.

`Router(config-if)# ip address ?`
`  A.B.C.D  IP address`
`  dhcp  IP Address negotiated via DHCP`
`  pool  IP Address autoconfigured from a local DHCP pool`
`Router(config-if)# ip address`
	Enter the command that you want to configure for the interface. This example uses the **ip address** command.
	Enter **?** to display what you must enter next on the command line. In this example, you must enter an IP address or the **negotiated** keyword.
	A carriage return (<cr>) is not displayed; therefore, you must enter additional keywords or arguments to complete the command.

`Router(config-if)# ip address 172.16.0.1 ?`
`  A.B.C.D  IP subnet mask`
`Router(config-if)# ip address 172.16.0.1`
	Enter the keyword or argument that you want to use. This example uses the 172.16.0.1 IP address.
	Enter **?** to display what you must enter next on the command line. In this example, you must enter an IP subnet mask.
	A <cr> is not displayed; therefore, you must enter additional keywords or arguments to complete the command.

Command	Comment

`Router(config-if)# ip address 172.16.0.1 255.255.255.0 ?`
`  secondary  Make this IP address a secondary address`
`  <cr>`
`Router(config-if)# ip address 172.16.0.1 255.255.255.0`
	Enter the IP subnet mask. This example uses the 255.255.255.0 IP subnet mask.
	Enter **?** to display what you must enter next on the command line. In this example, you can enter the **secondary** keyword, or you can press **Enter**.
	A <cr> is displayed; you can press **Enter** to complete the command, or you can enter another keyword.

`Router(config-if)# ip address 172.16.0.1 255.255.255.0` `Router(config-if)#`	In this example, **Enter** is pressed to complete the command.

Almost every configuration command has a **no** form. In general, use the **no** form to disable a function. Use the command without the **no** keyword to re-enable a disabled function or to enable a function that is disabled by default. For example, IP routing is enabled by default. To disable IP routing, use the **no ip routing** command;

to re-enable IP routing, use the **ip routing** command. The Cisco IOS XE software command reference publications provide the complete syntax for the configuration commands and describe what the **no** form of a command does.

Many CLI commands also have a **default** form. By issuing the command **default** *command-name* , you can configure the command to its default setting. The Cisco IOS XE software command reference publications describe the function of the **default** form of the command when the **default** form performs a different function than the plain and **no** forms of the command. To see what default commands are available on your system, enter **default ?** in the appropriate command mode.

Use the **copy running-config startup-config** command to save your configuration changes to the startup configuration so that the changes will not be lost if the software reloads or a power outage occurs. For example:

```
Router# copy running-config startup-config
Building configuration...
```

It might take a minute or two to save the configuration. After the configuration has been saved, the following output appears:

```
[OK]
Router#
```

This task saves the configuration to NVRAM.

On the Cisco CSR 1000v and ISRv, the startup configuration file is stored in the NVRAM partition. As a matter of routine maintenance on any Cisco router, users should backup the startup configuration file by copying the startup configuration file from NVRAM onto one of the router's other file systems and, additionally, onto a network server. Backing up the startup configuration file provides an easy method of recovering the startup configuration file in the event the startup configuration file in NVRAM becomes unusable for any reason.

The **copy** command can be used to backup startup configuration files. The following examples show the startup configuration file in NVRAM being backed up:

Example 1: Copying a Startup Configuration File to Bootflash

```
Router# dir bootflash:
Directory of bootflash:/
   11      drwx    16384     Jan 24 2012 04:53:55 -05:00     lost+found
   12      -rw-    289243620 Jan 24 2012 04:54:55 -05:00
   308257  drwx    4096      Jan 24 2012 04:57:06 -05:00    core
   876097  drwx    4096      Jan 24 2012 04:57:07 -05:00   .prst_sync
   63277  drwx   4096     Jan 24 2012 04:57:10 -05:00 .rollback_timer   13
      -rw-       0      Jan 24 2012 04:57:19 -05:00    tracelogs.
csr1000v-adventerprisek9.2012-01-23_12.39.SSA.bin
Router# copy nvram:startup-config bootflash:
Destination filename [startup-config]?
3517 bytes copied in 0.647 secs (5436 bytes/sec)
Directory of bootflash:/
   11      drwx    16384     Jan 24 2012 04:53:55 -05:00    lost+found
   12      -rw-    289243620 Jan 24 2012 04:54:55 -05:00
   308257  drwx    4096      Jan 24 2012 04:57:06 -05:00    core
   876097  drwx    4096      Jan 24 2012 04:57:07 -05:00   .prst_sync
   632737 drwx   4096     Jan 24 2012 04:57:10 -05:00 .rollback_timer   13
      -rw-       0      Jan 24 2012 04:57:19 -05:00    tracelogs.
csr1000v-adventerprisek9.2012-01-23_12.39.SSA.bin
   14 -rw-   7516       Jul 2 2012 15:01:39 -07:00     startup-config
```

Example 2: Copying a Startup Configuration File to a TFTP Server

```
Router# copy bootflash:startup-config tftp:
Address or name of remote host []? 172.17.16.81
Destination filename [pe24_asr-1002-confg]? /auto/tftp-users/user/startup-config
!!
3517 bytes copied in 0.122 secs (28828 bytes/sec)
```

For more detailed information on managing configuration files, see the " Managing Configuration Files" section in the [see "http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/fundamentals/configuration/xe-3s/fundamentals-xe-3s-book.html">Configuration Fundamentals Configuration Guide, Cisco IOS XE Release 3S].

NVRAM File Security

The Cisco CSR 1000v and ISRv encrypt some of the disk partitions internal to the VM to provide extra security around sensitive data that may be stored on the routers. For example, information in NVRAM is encrypted so that it is not visible to administrative entities with access to the physical hard disk upon which the Cisco CSR 1000v is stored.

You can search and filter the output of **show** and **more** commands. This functionality is useful if you need to sort through large amounts of output or if you want to exclude output that you need not see.

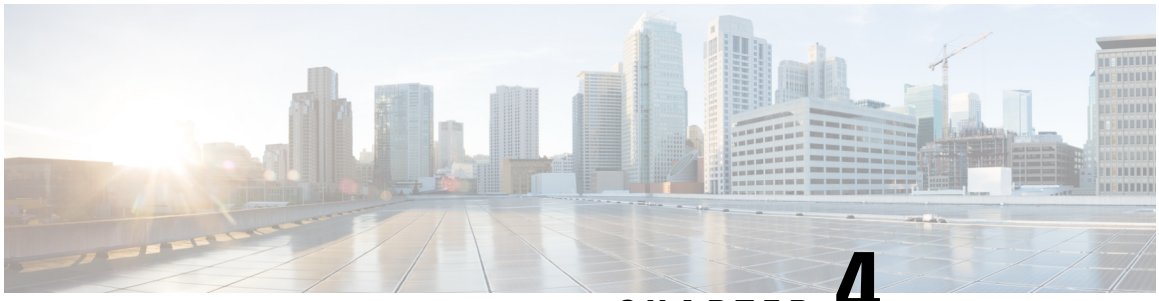
To use this functionality, enter a **show** or **more** command followed by the "pipe" character (|); one of the keywords **begin**, **include**, or **exclude**; and a regular expression on which you want to search or filter (the expression is case sensitive):

show*command* {**append** | | **begin** | | **exclude** | | **exclude** | | **include** | | **redirect** | | **section** | | **tee**}*regular-expression*

show *command* | {**append** | **begin** | **exclude | include** | **redirect** | **section** | **tee**} *regular-expression*

The output matches certain lines of information in the configuration file.

To power off a Cisco CSR 1000v, you must power off the VM upon which the router is installed. For information about powering off the VM, see your VM vendor documentation.



CHAPTER 4

Installation Overview

Introduction

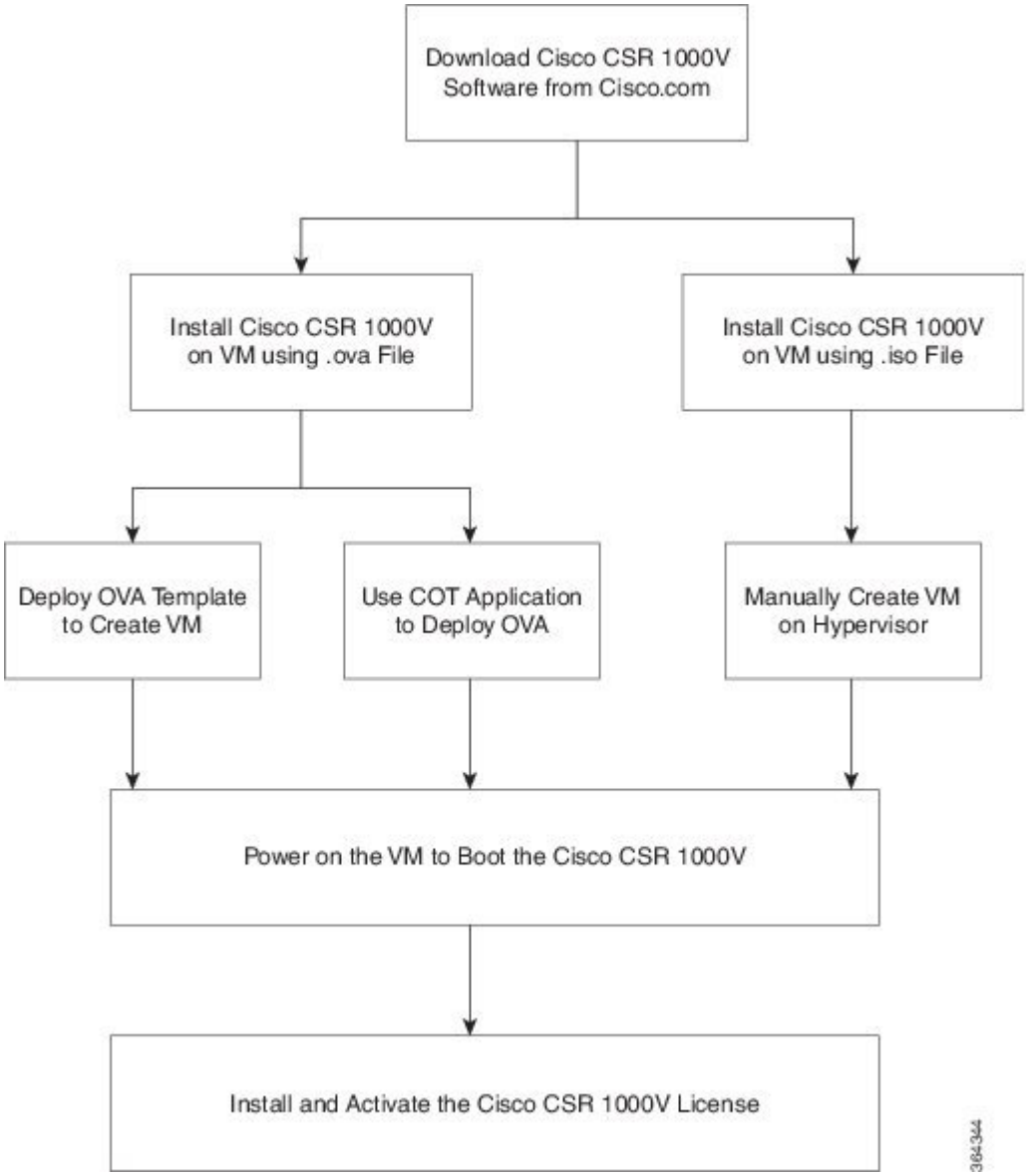
Cisco hardware routers are normally shipped with the Cisco IOS XE software pre-installed. Because the Cisco CSR 1000v Series Cloud Services Router is not hardware-based, you must download the Cisco IOS XE software from Cisco.com and install it directly onto the virtual machine. However, as part of the initial installation process, you must first provision the attributes of the VM so that the Cisco CSR 1000v software can install and boot.

Note For information about deploying the Cisco CSR 1000v in an Amazon Web Services environment, seehttp://www.cisco.com/c/en/us/td/docs/routers/csr1000/software/aws/csraws.html.

Note For information about deploying the Cisco CSR 1000v in a Microsoft Azure environment, see https://www.cisco.com/c/en/us/td/docs/routers/csr1000/software/azu/b_csr1000config-azure.html.

The figure below ("Cisco CSR 1000v Installation Task Workflow") shows the high-level tasks required to install the Cisco CSR 1000v on the VM. The different installation options are dependent on the hypervisor being used. See the following sections for more information.

Figure 2: Cisco CSR 1000v Installation Task Workflow



Virtual Machine Processing Resources

The Cisco CSR1000V is a low-latency application and might not function properly when the processing resources on the host side are over subscribed. By default, most hypervisors support overcommitting the processing resources. However, for Cisco CSR1000V, if you oversubscribe and do not schedule the virtual CPUs (vCPUs) reliably, you could experience packet processing drops, error messages, or system outages.

The Cisco CSR1000V vCPUs must be scheduled by the host hypervisor to run on real physical cores. Each hypervisor has various controls that influence the scheduling of the vCPUs to the physical cores. As a best practice, Cisco recommends that you to use a ratio of 1:1 for the vCPUs to real physical cores.

For detailed information on virtual machine processing resources, see the respective hypervisor tuning guides provided by the hypervisor. Additionally, you can refer to the appropriate hypervisor sections in this guide that describe the possible settings to increase the performance and improve the overall system determinism.

Obtaining the Cisco CSR 1000v VM Image

SUMMARY STEPS

1. Go to the Cisco Cloud Services Router 1000V Series product page: https://www.cisco.com/c/en/us/support/routers/cloud-services-router-1000v-series/tsd-products-support-series-home.html.
2. Click **Download Software** .
3. Select the router model (Cloud Services Router 1000v).
4. Click **IOS XE Software**. The recommended Cisco IOS XE release is selected by default.
5. In the list of available images, click **Download Now** or **Add to Cart**. Follow the instructions for downloading the software.

DETAILED STEPS

Procedure

Step 1	Go to the Cisco Cloud Services Router 1000V Series product page: https://www.cisco.com/c/en/us/support/routers/cloud-services-router-1000v-series/tsd-products-support-series-home.html.
Step 2	Click **Download Software** .
Step 3	Select the router model (Cloud Services Router 1000v).
Step 4	Click **IOS XE Software**. The recommended Cisco IOS XE release is selected by default.
Step 5	In the list of available images, click **Download Now** or **Add to Cart**. Follow the instructions for downloading the software.

Cisco CSR 1000v Installation Files

The following software images are available for installing the Cisco CSR 1000v on the supported hypervisors.

- .ova

Used for deploying the OVA template on the VM (in TAR format)

- .iso

Used for installing the software image on the VM (requires manually creating the VM)

- .qcow2

Used for installing the software image in KVM OpenStack environments.

- .run

(Cisco IOS XE 3.16 and later, and Cisco IOS XE Denali 16.3.1 and later) Self-installing image used for installation in a KVM environment.

- .bin

These images are used for upgrading and downgrading the software only. For more information, see Prerequisites for the Software Upgrade Process, on page 237 and subsequent sections.

Note (Cisco IOS XE Everest 16.5 and later) On AWS, you can use the Cisco CSR 1000v .bin file to upgrade the version of Cisco CSR 1000v, without having to recreate AWS EC2 instance from a new AMI. This inline upgrade process is not yet available on Microsoft Azure.

Note (Cisco IOS XE Everest 16.4 and earlier) You cannot use the Cisco CSR 1000v .bin file to upgrade AMIs obtained from Amazon Web Services. You must create a new AMI instance and migrate your configuration and licenses.

Cisco CSR 1000v Installation Options

Cisco CSR 1000v supports the following installation options:

- Deploy the OVA template on the VM.

  Uses the .ova file. This template creates a VM using recommended preset values. See Deploying the Cisco CSR 1000v OVA to the VM using vSphere, on page 79 and Deploying the Cisco CSR 1000v OVA to the VM using COT, on page 88.

  The .ova file can be used only for first-time installation. It cannot be used for upgrading the Cisco IOS XE software version.

- Deploy the .ova file on the VM using the Common OVF Tool (COT).

  The COT application is included in the file package. However, to ensure that you are using the latest version of COT, download COT directly from the GitHub site:

  https://github.com/glennmatthews/cot/blob/master/README.md

  Using the COT application, you can customize the VM values and easily deploy the custom VM as part of the Cisco CSR 1000v installation process. See Editing the Basic Properties of Cisco CSR 1000v using vSphere, on page 85.

Note The COT application is recommended in place of the BDEO tool, which is used in the early releases of Cisco IOS XE.

- Manually configure the VM using the .iso file.

  Uses the .iso file. You can install the .iso file on your host and manually create the VM using your hypervisor software. For example, if you are installing the Cisco CSR 1000v on VMware, you would

install the .iso file on the VMware ESXi host, and manually create the VM using the vSphere GUI. See the following sections:

- Manually Creating the Cisco CSR 1000v VM Using the .iso File (VMware ESXi), on page 95
- Manually Creating the Cisco CSR 1000v VM Using the .iso File (Citrix XenServer), on page 107
- Creating the Cisco CSR 1000v VM Using virt-install with ISO Image, on page 118
- Prerequisites for Manually Creating the CSR 1000v VM using the .iso File, on page 137

- Create the Cisco CSR 1000v instance in KVM using OpenStack

Uses the .qcow2 file. The qcow2 (QEMU Copy on Write) image format is used to create the Cisco CSR 1000v tenant in the KVM OpenStack cloud environment. See Selecting a Cisco CSR 1000v Installation Image, on page 120 onwards.

BDEO Tool

The Cisco Build, Deploy, Execute OVF (BDEO) tool is included in the OVA package. In past releases, this tool was recommended for Cisco CSR 1000v installation. The tool is **no longer recommended**, but is included in the package for unusual installation circumstances. You should use the COT application instead of the BDEO tool.

Upgrading Cisco IOS XE Software

For information about upgrading the Cisco IOS XE software, see Prerequisites for the Software Upgrade Process, on page 237and subsequent sections.

Installation Options and Requirements

The following table lists the installation options for the supported hypervisors and the minimum Cisco IOS XE software release required.

Table 16: Cisco CSR 1000v Supported Installation Options

Installation Option	VMware ESXi	Citrix XenServer	KVM	Microsoft Hyper-V
Deploy OVA Template Using OVA Wizard	Cisco IOS XE 3.9S and later	Not supported	Not supported	Not supported
Deploy OVA Using COT	Cisco IOS XE 3.9S and later	Not supported	Not supported	Not supported
Manually Configure VM Using .iso File	Cisco IOS XE 3.9S and later. (Cisco IOS XE Denali 16.2 is not supported by CSR 1000v.)	Cisco IOS XE 3.10S and later. (Cisco IOS XE Denali 16.2 is not supported by CSR 1000v.)	Cisco IOS XE 3.10S and later. (Cisco IOS XE Denali 16.2 is not supported by CSR 1000v.)	Cisco IOS XE 3.12S and later. (Cisco IOS XE Denali 16.2 is not supported by CSR 1000v.)
Create the KVM instance on OpenStack Using .qcow2 File	NA	NA	Cisco IOS XE 3.12S and later. (Cisco IOS XE Denali 16.2 is not supported by CSR 1000v.)	NA

Note When a device is in the installation mode, formatting of the boot drive, bootflash/flash is not recommended. Formatting is blocked to ensure stability of the running image and to avoid any impact to upgrade of the software.

Guidelines and Limitations

Be aware of the following general guidelines and restrictions before installing the Cisco CSR1000V in your network:

- The Cisco CSR1000V may properly function within a nested VM, but this is not tested nor supported.
- If the hypervisor does not support vNIC Hot Add/Remove, do not make any changes to the VM hardware (memory, CPUs, hard drive size, and so on) while the VM is powered on.
- (Cisco IOS XE Release 3.11S and later) GigabitEthernet0 interface is no longer available. You can designate any interface as the management interface.
- (Cisco IOS XE Release 3.10S and earlier) GigabitEthernet0 interface is the default management port and cannot be changed.
- You can access the Cisco IOS XE CLI either through the virtual VGA console or the console on the virtual serial port. The console can be selected from GRUB mode during the first-time installation, or it can be changed using the Cisco IOS XE **platform console** command after the router boots. For more information, see Booting the Cisco CSR 1000v as the VM, on page 143.
- If you deploy a Cisco CSR1000V 16.12 image, you can downgrade to the 16.11.x release only, and not any of the previous releases. When you attempt to downgrade directly to an earlier release, the device boots back with the 16.12 image. If you deploy a Cisco CSR1000V 16.10 or a 16.9 image, and then upgrade to 16.12, you can downgrade the Cisco CSR1000V instance to 16.10.x or 16.9.x releases.

Note Some hypervisors may not support serial console access. Verify support using your hypervisor documentation.

ROMMON and the Cisco CSR 1000v

The Cisco CSR 1000v, which is software-based, does not include a ROMMON image. This differs from many Cisco hardware-based routers. During the initial bootloader process, the installation script creates a clean version of the Cisco CSR 1000v software image known as the Golden Image and places it in a non-accessible partition. This clean version can be used if the software image is not working properly or is not bootable.

Note that although the Cisco CSR 1000v does not include ROMMON, the platform does include a GNU GRand Unified Bootloader (GRUB)-based bootloader. The GRUB function on the Cisco CSR 1000v provides more limited functionality compared to the ROMMON available on other Cisco platforms.

Note that although ROMMON is not present on the Cisco CSR 1000v, some Cisco IOS XE commands such as **show version** may show references to ROMMON in the command output.

Note After the Cisco CSR 1000v completes the first-time installation, you can configure the router to automatically enter GRUB mode when the router is booted. For more information, see Activating Cisco CSR 1000v Licenses, on page 162 and subsequent licensing sections.

CSR and ISRv - VNF Secure Boot

The secure boot feature prevents malicious software applications and unauthorized operating systems from loading into the system during the system startup process. If the secure boot feature is enabled, only the authorized software applications boots up from the device. This feature ensures that the software applications that boot up on the device are certified by Cisco. A secure compute system ensures that the intended software on the system runs without malware or tampered software. The UEFI (Unified Extensible Firmware Interface) specification defines a secure boot methodology that prevents loading software which is not signed with an acceptable digital signature.

To display the system boot mode and the bootloader version use **show platform software system boot** command.

```
Router#show platform software system boot
Boot mode: EFI
Bootloader version: 2.0
```

Restrictions

- The following secure boot environments are supported:
  - ESXi version 6.5 or higher
  - KVM RHEL 7.5 using open stack license
  - NFVIS release 3.11 or later
- Only EFI firmware modes support the secure boot.
- This feature is supported on VM created in Cisco IOS XE Gibraltar 16.12 or later releases.

  GRUB2 and new disk partition layout available for Cisco IOS XE Gibraltar 16.12 or later releases.

Note VMs created before Cisco IOS XE Gibraltar 16.12 release supports GRUB and BIOS mode and does not upgrade to GRUB2.

Note Each hypervisor has a unique process to enable secure boot for the guest VMs. Refer to hypervisor specific documentation to enable secure boot. A set of high-level hypervisor specific steps to enable secure boot are mentioned below.

ESXi Secure Boot Setup

- Create VM using ESXi 6.5 or later version using VM version 13 or greater. To choose the EFI firmware mode, navigate through **VM Options** > **Boot Options** > **Firmware** > **EFI**.
- Power down the VM after the initial boot and IOS prompt is complete.
- Enable the EFI secure boot in **Edit Settings** > **VM Options** > **Boot Options** > **Secure Boot**.
- Power up VM and the VNF boots up securely.

KVM Secure Boot Setup

- Create the VM.
- Power down the VM after the VM is created and VNF IOS prompt is complete.
- Install PK, KEK, and db certificates from the **EFI Firmware** menu and reset.

  To create the custom keys, see Custom Keys for Secure boot. For db certificates, see MicCorUEFCA2011_2011-06-27.crt and MicWinProPCA2011_2011-10-19.crt.
- Secure boot the VM.

NFVIS Secure Boot Setup

- Upgrade to NFVIS 3.11 release or later.
- Register an ISRv EFI tarball with the NFVIS repository.
- Create a VM using the registered EFI image.
- Secure boot the VM.

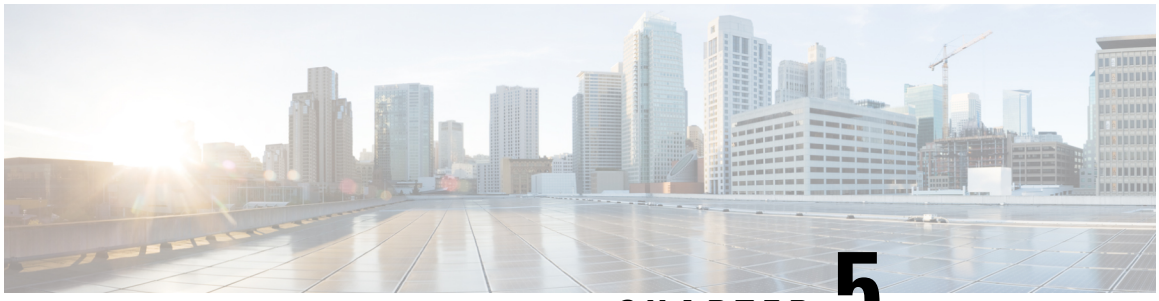
Where to Go Next

See the information in the sections below, about installing the Cisco CSR 1000v in different hypervisor environments:

- VMware ESXi Support Information, on page 73
- Microsoft Hyper-V Support Information, on page 135
- Citrix XenServer Support Information, on page 105
- Kernel Virtual Machine Support Information, on page 111

Note For information about deploying the Cisco CSR 1000v in an Amazon Web Services environment, see the Cisco CSR 1000V Series Cloud Services Router Deployment Guide for Amazon Web Services.

Note For information about deploying the Cisco CSR 1000v in a Microsoft Azure environment, see the Cisco CSR 1000v Deployment Guide for Microsoft Azure.



CHAPTER 5

Installing the Cisco CSR 1000v in VMware ESXi Environments

VMware ESXi Support Information

This chapter contains information about VMware tools/software and the VM requirements for Cisco CSR 1000v / Cisco IOS XE software.

The Cisco CSR 1000v can run on the VMware ESXi hypervisor. VMware ESXi runs on PCs with x86-based CPUs. You can use the same hypervisor to run several VMs.

VMware vSphere Web Client is a web application that runs on a PC and accesses VMware vCenter Server. You can use VMware vSphere Web Client software to create, configure, and manage VMs on the vCenter Server and to start/stop the Cisco CSR 1000v. The Cisco CSR 1000v boots from a virtual disk located on the data store.

Note If you upgrade VMware ESXi, and ESXi contains an existing Cisco CSR 1000v, the interfaces of the CSR 1000v may be renamed. For example, GigabitEthernet1 may appear as GigabitEthernet4. To recover the original interface names, perform the following two Cisco IOS XE configuration commands from the console or terminal of the CSR 1000v, immediately after upgrading the VMware ESXi hypervisor:

clear platform software vnic nvtable

reload

Caution Oversubscription of host resources can lead to a reduction of performance and your instance could become instable. We recommend that you follow the guidelines and the best practices for your host hypervisor

Note Only ESXi hypervisor supports the FIPS mode. Ensure that you have configured the virtual machine properly to allow entropy gathering when in FIPS mode. Failing to configure ESXi properly or using another hypervisor results in the device crashing. This is applicable only for CSR 1000v release 16.9.x. The 16.10 image and later allows entropy when you use any supported hypervisor.

To find out more about installing VMware vSphere products, see VMware product documentation .

VMware Requirements

The following table specifies the supported VMware tools by Cisco CSR 1000V using Cisco IOS XE Amsterdam 17.x releases:

Cisco IOS XE Release	vSphere Web Client	vCenter Server
Cisco IOS XE 17.3.x releases	The 6.7 and 6.5 versions of the VMware vSphere Web Client are supported.	VMware ESXi 6.7 and EsXi 6.5
Cisco IOS XE 17.1.x and 17.2.x releases	The 6.5 Update 1 and 6.5 Update 2 versions of the VMware vSphere Web Client are supported.	VMware ESXi 6.5 Update 2
Cisco IOS XE 16.12 release	The 6.5 Update 1 and 6.5 Update 2 versions of the VMware vSphere Web Client are supported.	VMware ESXi 6.5 Update 2

These versions have been fully tested and meet performance benchmarks.

VMware vCenter—installation tool.

VMware vSwitch—standard or distributed vSwitches are supported.

Hard Drive—only a single hard disk drive is supported. Multiple hard disk drives on a VM are not supported.

Virtual Disk—a 8 GB virtual disk is supported.

vCPUs—the following vCPU configurations are supported:

Note The required vCPU configuration depends on the throughput license and technology package installed. For more information, see the data sheet for your release.

- 1 vCPU: requires minimum 4 GB RAM allocation
- 2 vCPUs: requires minimum 4 GB RAM allocation

- 4 vCPUs: requires minimum 4 GB RAM allocation
- 8 vCPUs: requires minimum 4 GB RAM allocation

Virtual CPU core—one virtual CPU core is required. This needs a 64-bit processor with Virtualization Technology (VT) enabled in the BIOS setup of the host machine.

Virtual hard disk space—minimum size of 8 GB.

Virtual Network Interface Cards (vNICs)—Three or more vNICs (max. 10)—VMXNET3 or i40evf.

A default video, SCSI controller set is required, and an installed virtual CD/DVD drive.

Note For VMware requirements—Cisco IOS XE 3.x, see VMware Requirements—Cisco IOS XE 3.x, on page 98 and VMware VM Requirements—Cisco IOS XE 3.x, on page 100.

Supported VMware Features and Operations

VMware supports various features and operations that allow you to manage your virtual applications and perform operations such as cloning, migration, shutdown and resume.

Some of these operations cause the runtime state of the VM to be saved and then restored upon restarting. If the runtime state includes traffic-related state, then on resumption or replaying the runtime state, additional errors, statistics, or messages are displayed on the user console. If the saved state is just configuration driven, you can use these features and operations without a problem.

The table "Supported VMware Features and Operations: Storage Options (for Both vCenter Server and vSphere Client)" lists the VMware features and operations that are supported on the Cisco CSR 1000v. For more information about VMware features and operations, see the VMware Documentation .

The following VMware features and operations are not supported in all versions of the Cisco CSR 1000v, but can still be used or performed on non-supported versions at the risk of encountering dropped packets, dropped connections, and other error statistics:

- Distributed Resource Scheduling (DRS)
- Fault Tolerance
- Resume
- Snapshot
- Suspend

See the following sections for more information.

General Features (vCenter Server)

Table 17: Supported VMware Features and Operations: General Features (for vCenter Server Only)

Supported Entities	First Supported Cisco CSR 1000v Release	Description
Cloning	Cisco IOS XE Release 3.9S	Enables cloning a virtual machine or template, or cloning a virtual machine to a template.
Migrating	Cisco IOS XE Release 3.9S	The entire state of the virtual machine as well as its configuration file, if necessary, is moved to the new host even while the data storage remains in the same location on shared storage.
vMotion	Cisco IOS XE Release 3.9S	Enables moving the VM from one physical server to another while the VM remains active.
Template	Cisco IOS XE Release 3.9S	Uses templates to create new virtual machines by cloning the template as a virtual machine.

Operations (for vCenter Server and vSphere Web Client)

Table 18: Supported VMware Features and Operations: Operations (for vCenter Server and vSphere Client)

Supported Entities	First Supported Cisco CSR 1000v Release	Description
Power On	Cisco IOS XE Release 3.9S	Powers on the virtual machine and boots the guest operating system if the guest operating system is installed.
Power Off	Cisco IOS XE Release 3.9S	Stops the virtual machine until it is powered back. The power off option performs a “hard” power off, which is analogous to pulling the power cable on a physical machine and always works.
Shut Down	Not supported.	Shut Down, or “soft” power off, leverages VMware Tools to perform a graceful shutdown of a guest operating system. In certain situations, such as when VMware Tools is not installed or the guest operating system is hung, shut down might not succeed and using the Power off option is necessary.
Suspend	Not supported	Suspends the virtual machine.
Reset/Restart	Cisco IOS XE Release 3.9S	Stops the virtual machine and restarts (reboots) it.
OVF Creation	Cisco IOS XE Release 3.9S	An OVF package consisting of several files in a directory captures the state of a virtual machine including disk files that are stored in a compressed format. You can export an OVF package to your local computer.
OVA Creation	Cisco IOS XE Release 3.9S	You can create a single OVA package file from the OVF package/template. The OVA can then be distributed more easily; for example, it may be downloaded from a website or moved via a USB key.

Table 19: Supported VMware Features and Operations: Networking Features

Supported Entities	First Supported Cisco CSR 1000v Release	Description
Custom MAC address	Cisco IOS XE Release 3.9S	From both vCenter Server and vSphere Client. Allows you to set up the MAC address manually for a virtual network adapter.
Distributed VSwitch	Cisco IOS XE Release 3.9S	From vCenter Server only. A vSphere distributed switch on a vCenter Server data center can handle networking traffic for all associated hosts on the data center.
Distributed Resources Scheduler	Cisco IOS XE Release 3.10S	Provides automatic load balancing across hosts.
NIC Load Balancing	Cisco IOS XE Release 3.9S	From both vCenter Server and vSphere Client. Load balancing and failover policies allow you to determine how network traffic is distributed between adapters and how to reroute traffic if an adapter fails.
NIC Teaming	Cisco IOS XE Release 3.9S	From both vCenter Server and vSphere Client. Allows you to set up an environment where each virtual switch connects to two uplink adapters that form a NIC team. The NIC teams can then either share the load of traffic between physical and virtual networks among some or all of its members, or provide passive failover in the event of a hardware failure or a network outage. **Note** NIC Teaming can cause a large number of ARP packets to flood the Cisco CSR 1000v and overload the CPU. To avoid this situation, reduce the number of ARP packets and implement NIC Teaming as Active-Standby rather than Active-Active.
vSwitch	Cisco IOS XE Release 3.9S	From both vCenter Server and vSphere Client. A vSwitch is a virtualized version of a Layer 2 physical switch. A vSwitch can route traffic internally between virtual machines and link to external networks. You can use vSwitches to combine the bandwidth of multiple network adapters and balance communications traffic among them. You can also configure a vSwitch to handle a physical NIC failover.

High Availability

Note Cisco IOS-based High Availability is not supported by the Cisco CSR 1000v. High Availability is supported on the VM host only.

Table 20: Supported VMware Features and Operations: High Availability

Supported Entities	First Supported Cisco CSR 1000v Release	Description
VM-Level High Availability	Cisco IOS XE Release 3.9S	To monitor operating system failures, VM-Level High Availability monitors heartbeat information in the VMware High Availability cluster. Failures are detected when no heartbeat is received from a given virtual machine within a user-specified time interval. VM-Level High Availability is enabled by creating a resource pool of VMs using VMware vCenter Server.
Host-Level High Availability	Cisco IOS XE Release 3.9S	To monitor physical servers, an agent on each server maintains a heartbeat with the other servers in the resource pool such that a loss of heartbeat automatically initiates the restart of all affected virtual machines on other servers in the resource pool. Host-Level High Availability is enabled by creating a resource pool of servers or hosts, and enabling high availability in vSphere.
Fault Tolerance	Cisco IOS XE Release 3.10S	Using high availability, fault tolerance is enabled on the ESXi host. When you enable fault tolerance on the VM running the Cisco CSR 1000v, a secondary VM on another host in the cluster is created. If the primary host goes down, then the VM on the secondary host will take over as the primary VM for the Cisco CSR 1000v.

Storage Options (for vCenter Server and vSphere Web Client)

Table 21: Supported VMware Features and Operations: Storage Options (for Both vCenter Server and vSphere Client)

Supported Entities	First Supported Cisco CSR 1000v Release	Description
Storage Options (for both vCenter Server and vSphere Client)		
Local Storage	Cisco IOS XE Release 3.9S	Local storage is in the internal hard disks located inside your ESXi host. Local storage devices do not support sharing across multiple hosts. A datastore on a local storage device can be accessed by only one host.

Supported Entities	First Supported Cisco CSR 1000v Release	Description
External Storage Target	Cisco IOS XE Release 3.9S	You can deploy the Cisco CSR 1000v on external storage, that is, a Storage Area Network (SAN).
Mount or Pass Through of USB Storage	Cisco IOS XE Release 3.9S	You can connect USB sticks to the Cisco CSR 1000v and use them as storage devices. In ESXi, you need to add a USB controller and then assign the disk devices to the Cisco CSR 1000v. • Cisco CSR 1000v supports USB disk hot-plug. • You can use only two USB disk hot-plug devices at a time. • USB hub is not supported.

Deploying the Cisco CSR 1000v OVA to the VM

Deploying the Cisco CSR 1000v OVA to the VM

You can use the provided CSR 1000v OVA file package to deploy the Cisco CSR 1000v to the VM. The OVA package includes an OVF file that contains a default VM configuration based on a Cisco IOS XE release and the supported hypervisor. (See the “Guidelines and Limitations” section of the installation configuration that is included in the OVA file.)

The OVA can be deployed using VMware vSphere or COT (Common OVF Tool).

• Deploying the Cisco CSR 1000v OVA to the VM using vSphere, on page 79
• Deploying the Cisco CSR 1000v OVA to the VM using COT, on page 88

Note The Citrix XenServer, KVM and Microsoft Hyper-V implementations do not support deploying the VM using the .ova file. You must manually install the VM using the .iso file.

Deploying the Cisco CSR 1000v OVA to the VM using vSphere

Deploying the Cisco CSR 1000v OVA to the VM using vSphere

You can use the provided CSR 1000v OVA file package to deploy the Cisco CSR 1000v to the VM. The OVA package includes an OVF file that contains a default VM configuration based on a Cisco IOS XE release and the supported hypervisor. (See the “Guidelines and Limitations” section of the installation configuration that is included in the OVA file.)

Note The Citrix XenServer, KVM and Microsoft Hyper-V implementations do not support deploying the VM using the .ova file. You must manually install the VM using the .iso file.

- Restrictions and Requirements, on page 80
- Deploying the Cisco CSR 1000v OVA to the VM, on page 79
- Editing the Basic Properties of Cisco CSR 1000v using vSphere, on page 85
- Editing the Custom Properties of Cisco CSR 1000v using vSphere, on page 87

Restrictions and Requirements

The following restrictions apply when deploying the OVA package to the VM:

- (Cisco IOS XE Releases 3.10S and 3.11S) The OVA package only creates a VM with 4 virtual CPUs. To change to the 1 or 2 virtual CPU configuration, first deploy the OVA template, and then use vSphere to change the virtual CPU configuration and the required RAM allocation.

If the virtual CPU configuration is changed, the Cisco CSR 1000v must be rebooted. Changing the RAM allocation does not require rebooting the Cisco CSR 1000v. Beginning with Cisco IOS XE 3.12S, the OVA package provides an option to select the virtual CPU configuration.

- When deploying the OVA, the VM requires two virtual CD/DVD drives, one for the OVF environment file and one for the .iso file.
- When HSRP or IPV6 link local runs in regular VMware environments and the vSwitch has a redundant uplink of 2 vmnics connected to the same L2 segment, the VMswitch physical uplinks will be connected in a loop or both uplinks will be connected to a switched infrastructure. This process leads to l2 flooding. For more information on addressing this scenario from the VMware side, refer to this link: https://kb.vmware.com/s/article/59235.

Deploying the OVA to the VM

Perform the following steps in VMware vSphere Client:

SUMMARY STEPS

1. Log in to the VMware vSphere Client.
2. From the vSphere Client Menu Bar, choose File > Deploy OVF Template.
3. In the OVA Wizard, point the source to the Cisco CSR 1000v OVA to be deployed. Click **Next**.
4. Under Name and Inventory Location, specify the name for the VM and click **Next**.
5. (Cisco IOS XE Release 3.12S and later): Under Deployment Configuration, select the desired hardware configuration profile from the drop-down menu and click **Next**.
6. Under Storage, select the Datastore to use for the VM. Click **Next**.
7. Under Disk Format, select the disk format option:
8. Under Network Mapping, allocate one or more virtual network interface card (vNIC) on the destination network using the drop-down list. The options for mapping the vNICs differ depending on the release version:
9. Configure the properties for the VM.

10. Select **Power on after deployment** to automatically power on the VM.
11. Click **Finish** to deploy the OVA.

DETAILED STEPS

Procedure

Step 1 Log in to the VMware vSphere Client.

Step 2 From the vSphere Client Menu Bar, choose File > Deploy OVF Template.

Step 3 In the OVA Wizard, point the source to the Cisco CSR 1000v OVA to be deployed. Click **Next**.

OVF Template Details appears, showing information about the OVA. Click **Next**.

Step 4 Under Name and Inventory Location, specify the name for the VM and click **Next**.

Step 5 (Cisco IOS XE Release 3.12S and later): Under Deployment Configuration, select the desired hardware configuration profile from the drop-down menu and click **Next**.

Step 6 Under Storage, select the Datastore to use for the VM. Click **Next**.

Step 7 Under Disk Format, select the disk format option:

- Thick Provision Lazy Zeroed
- Thick Provision Eager Zeroed

Note
The Thin Provision option is not supported. The Thick Provision Eager Zeroed option takes longer to install but provides better performance.

Click Next.

Step 8 Under Network Mapping, allocate one or more virtual network interface card (vNIC) on the destination network using the drop-down list. The options for mapping the vNICs differ depending on the release version:

- (Cisco IOS XE Release 3.11S and later, and IOS XE Denali 16.2 and later): Select the network mappings for the 3 default vNICs created during the OVA deployment. You can choose which vNIC will map to the router's management interface when setting the bootstrap properties (see table "Bootstrap Properties for Cisco IOS XE Release 3.11S and Later" below).

  Note
  After you make any change to the bootstrap properties the system assumes that you are starting with a fresh VM. So when the VM restarts, all pre-existing networking configuration will have been removed..

- (Cisco IOS XE Release 3.10S and earlier) The vNIC allocated in this step is mapped to the GigabitEthernet0 management interface on the router.

Select the vNIC to connect at Power On. Click Next.

When the Cisco CSR 1000v installation using the OVA is complete, two additional vNICS are allocated. The Cisco CSR 1000v supports up to ten vNICs; additional vNICs must be manually created on the VM.

The Properties screen displays.

Step 9 Configure the properties for the VM.

Note
After you make any change to the bootstrap properties the system assumes that you are starting with a fresh VM. So when the VM restarts, all pre-existing networking configuration will have been removed.

The available properties differ depending on the Cisco IOS XE release that you are using. See the tables below for the OVA bootstrap properties for the relevant release of Cisco IOS XE.

Note
The bootstrap properties are optional when creating the VM. You can set these properties to easily provision the VM before starting it up.

Table 22: OVA Bootstrap Properties for Cisco IOS XE Release 3.11S and Later

Property	Description
Bootstrap Properties	
Console	(Cisco IOS XE 3.17S and later, and Denali 16.2 and later) Configures the console mode. Possible values: auto, virtual, serial
Login Username	Sets the login username for the router.
Login Password	Sets the login password for the router.
Management Interface	Designates the management interface for the Cisco CSR 1000v. The format must be GigabitEthernetx or GigabitEthernetx.xxx. **Note** The GigabitEthernet0 interface is no longer supported beginning in Cisco IOS XE Release 3.11S.
Management vLAN	Configures the dot1Q VLAN interface. Requires the management interface to be configured using the GigabitEthernetx.xxx format.
Management Interface IPv4 Address/Mask	Configures the IPv4 address and subnet mask for the management interface.
Management IPv4 Default Gateway	(Cisco IOS XE Release 3.11S through 3.17S, and Denali 16.2 and later) Configures the IPv4 management default gateway address. If using DHCP, enter “dhcp” in the field.
Management IPv4 Gateway	(Cisco IOS XE Release 3.12S through 3.17S, and Denali 16.2 and later) Configures the IPv4 management default gateway address. If using DHCP, enter “dhcp” in the field.
Management IPv4 Network	(Cisco IOS XE Release 3.12S through 3.17S, and Denali 16.2 and later) Configures the IPv4 Network (such as “192.168.2.0/24” or “192.168.2.0 255.255.255.0”) that the management gateway should route to. If a default route (0.0.0.0/0) is desired, this may be left blank.

Property	Description
Remote Management IPv4 Address	(Optional) Configures the IP address used for remote management of the Cisco CSR 1000v by the REST API or by the Cisco Prime Network Services Controller. The address must be in the same subnet as the management interface address. **Note** Beginning with Cisco IOS XE 3.13S, this option is not used if configuring the shared management interface to support REST API. See Introduction to REST API Configuration Options, on page 319.
PNSC IPv4 Address	Configures the IP address of the Cisco Prime Network Services Controller. This setting is used if you plan to remotely manage the Cisco CSR 1000v using the Cisco Prime Network Services Controller.
PNSC Agent Local Port	(Optional) Configures the Cisco Prime Network Services Controller service agent SSL port on the local Cisco CSR 1000v to receive policies from the service manager. This setting is used if you plan to remotely manage the Cisco CSR 1000v using the Cisco Prime Network Services Controller.
PNSC Shared Secret Key	Configures the Cisco Prime Network Services Controller shared secret key for the Cisco Prime Network Services Controller agent to set the SSL certificate from the controller. This setting is used if you plan to remotely manage the Cisco CSR 1000v using the Cisco Prime Network Services Controller.
Router name	Configures the hostname of the router.
Resource Template	(Cisco IOS XE 3.16S2 and later, and Denali 16.2 and later) Configures the Resource Template. Possible values: default, service_plane_medium, service_plane_heavy
Features	
Enable SCP Server	Enables the IOS SCP feature.
Enable SSH Login Enable SSH Login and Disable Telnet Login	(Enable SSH Login, Cisco IOS XE Release 3.11S) (Enable SSH Login and Disable Telnet Login, Cisco IOS XE Release 3.12S and later, and Denali 16.2 and later) Enables remote login using SSH and disables remote login via Telnet. Requires that the login username and password are set.
Additional Configuration Properties	
Enable Password	Configures the password for privileged (enable) access.
Domain Name	Configures the network domain name.
License Boot Level	(Cisco IOS XE 3.13S and later, and Denali 16.2 and later) Configures the license technology level that is available when the Cisco CSR 1000v boots.

Table 23: OVA Bootstrap Properties for Cisco IOS XE Release 3.9S and 3.10S

Property	Description
Bootstrap Properties	
Login Username	Sets the login username for the router.
Login Password	Sets the login password for the router.
Management IPv4 Address/Mask	Sets the management gateway address/mask in IPv4 format for the GigabitEthernet0 management interface.
Management IPv4 Default Gateway	Sets the default management gateway IP address in IPv4 format for the GigabitEthernet0 management interface.
Router name	Configures the hostname of the router.
Features	
Enable HTTP Server	(Cisco IOS XE Release 3.9S only) Enables an HTTP server for system configuration and administration via a web browser.
Enable HTTPS Server	(Cisco IOS XE Release 3.10S only) Enables an HTTPS server for system configuration and administration via a web browser. Required if using the REST API to perform system configuration. **Note** The HTTPS server is enabled by default beginning in Cisco IOS XE Release 3.11S. This field was removed.
Enable SSH Login	Enables remote login using SSH and disables remote login via Telnet. Requires that the login username and password are set.
Additional Configuration Properties	
Enable Password	Configures the password for privileged (enable) access.
Domain Name	Configures the network domain name.

When finished configuring the router properties, click Next. The Ready to Complete screen displays, showing the settings to be used when the OVA is deployed.

You can also configure advanced properties after the router boots.

Step 10 Select **Power on after deployment** to automatically power on the VM.

Step 11 Click **Finish** to deploy the OVA.

The OVA deploys the .iso file and, if the "Power on after deployment" setting is selected, automatically powers on the VM. Once the VM is powered on, the Cisco CSR 1000v begins the installation and boot process. If a bootstrap configuration file was included in the OVA, the router configuration will automatically be enabled.

See Booting the Cisco CSR 1000v and Accessing the Console, on page 143.

Editing the Basic Properties of Cisco CSR 1000v using vSphere

When deploying the OVA template, you have the option to set basic router properties using the vSphere GUI prior to booting, as described in Deploying the Cisco CSR 1000v OVA to the VM using vSphere, on page 79. You can also set custom properties matched to Cisco IOS XE CLI commands. See Editing the Custom Properties of Cisco CSR 1000v using vSphere, on page 87.

Note The functionality described in this chapter works only when using the vSphere GUI to connect to a vCenter server. If connecting directly to a host, these options are not available.

If the VM was manually created from the .iso file, then the vSphere GUI will not provide options to set basic router properties. However, you can still set custom properties as described in Editing the Custom Properties of Cisco CSR 1000v using vSphere, on page 87 . If you wish to do so, you will need to add a second virtual CD/DVD drive to the VM for vCenter to pass these properties into the VM.

To edit the vApp options to set basic Cisco CSR 1000v properties, do the following:

SUMMARY STEPS

1. In the vSphere GUI, select the Options tab.
2. Choose vApp Options > Properties.
3. Click on the Properties button.
4. Select the property to be edited and click Edit.
5. Once you have edited the property, click OK to close.

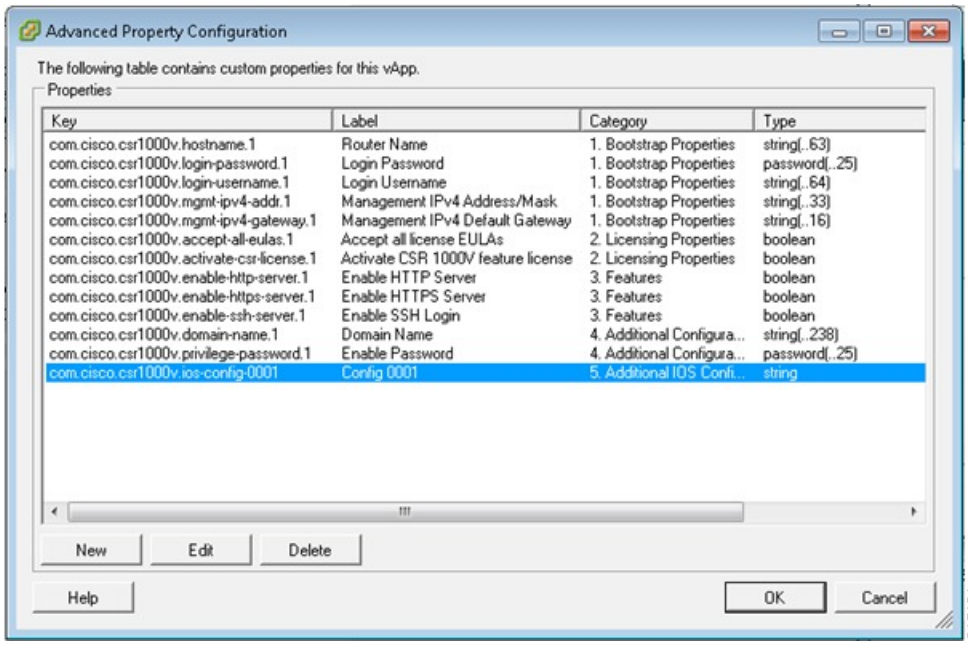
DETAILED STEPS

Procedure

Step 1 In the vSphere GUI, select the Options tab.

Step 2 Choose vApp Options > Properties.

Figure 3: vApp Advanced Options for Cisco CSR 1000v

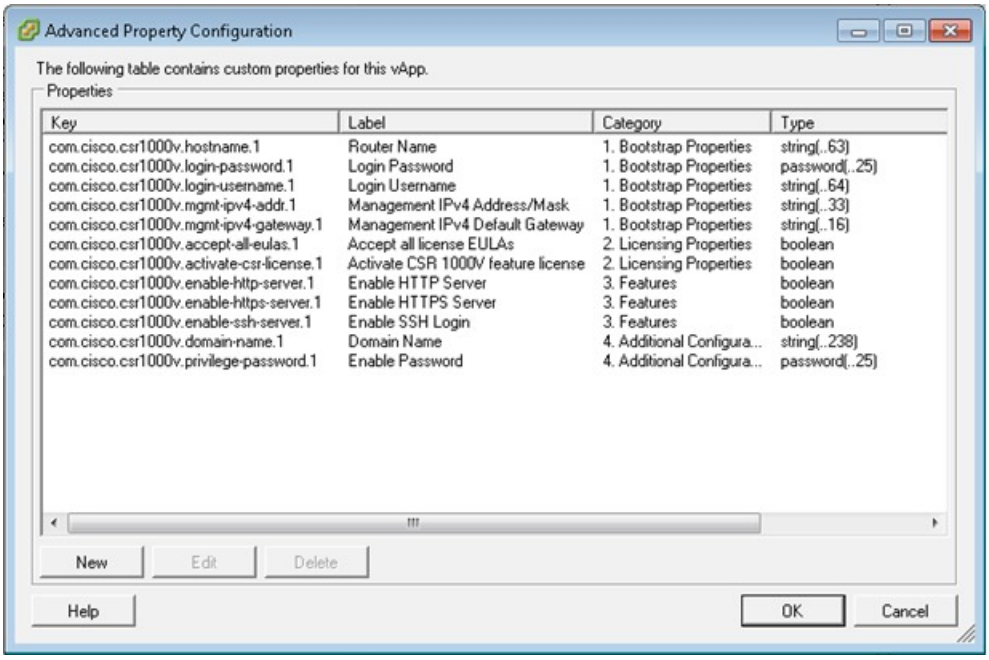


Step 3 Click on the Properties button.

A new window opens that provides access to the properties that can be edited. See the example Advanced Property Configuration Screen below.

Note
These properties can also be set using selected steps of the procedure described in Deploying the Cisco CSR 1000v OVA to the VM using vSphere, on page 79.

Figure 4: Cisco CSR 1000v Advanced Property Configuration Screen



See the tables in Deploying the Cisco CSR 1000v OVA to the VM using vSphere, on page 79 for the basic Cisco CSR 1000v properties that can be edited in the vSphere vApps GUI.

Step 4 Select the property to be edited and click Edit.

Step 5 Once you have edited the property, click OK to close.

Editing the Custom Properties of Cisco CSR 1000v using vSphere

You can add custom properties to the Cisco CSR 1000v based on Cisco IOS XE CLI commands using the vSphere GUI. You can add these properties either before or after you boot the Cisco CSR 1000v. If you set these custom properties after the Cisco CSR 1000v has booted, you will need to reload the router or power-cycle the VM for the properties settings to take effect.

To edit the vApp options to add custom Cisco CSR 1000v properties, do the following:

SUMMARY STEPS

1. In the vSphere GUI, select the Options tab.
2. Choose vApp Options > Advanced.
3. Click on the Properties button.
4. Click New to add a property.
5. Enter the information to create the new custom property based on a Cisco IOS XE CLI command:
6. When finished, click OK.
7. In the Advanced Property Configuration window, click OK.
8. Reboot the Cisco CSR 1000v.

DETAILED STEPS

Procedure

Step 1 In the vSphere GUI, select the Options tab.

Step 2 Choose vApp Options > Advanced.

The Advanced Property Configuration window appears.

Step 3 Click on the Properties button.

Step 4 Click New to add a property.

The Edit Property Settings window appears.

Step 5 Enter the information to create the new custom property based on a Cisco IOS XE CLI command:

Note
Before adding a custom property, make sure that the Cisco IOS XE command upon which it is based is supported on the Cisco CSR 1000v in your release.

a) (Optional) Enter the label. This is a descriptive string for the property.
b) Enter the class ID as "com.cisco.csr1000v".
c) Assign the property an ID of "ios-config-xxxx" where xxxx is a sequence number from 0001 to 9999 that determines the order in which the custom properties are applied.
d) (Optional) Enter a description for the property.
e) Enter the property type as "string". This is the only type supported.
f) Enter the default value as the Cisco IOS XE CLI command the custom property is based on.

Step 6 When finished, click OK.

Step 7 In the Advanced Property Configuration window, click OK.

Step 8 Reboot the Cisco CSR 1000v.

The router must reboot in order for the new or edited properties to take effect.

Deploying the Cisco CSR 1000v to the VM using COT

Deploying the Cisco CSR 1000v OVA to the VM using COT

You can use the provided CSR 1000v OVA file package to deploy the Cisco CSR 1000v to the VM. The OVA package includes an OVF file that contains a default VM configuration based on a Cisco IOS XE release and the supported hypervisor. (See the "Guidelines and Limitations" section of the installation configuration that is included in the OVA file.) The OVA can be deployed using VMware vSphere or COT (Common OVF Tool). This section describes how to deploy using the COT (Common OVF Tool).

The Common OVF Tool (COT) included in the Cisco CSR 1000v software package is a Linux-based application that enables you to create attributes for one or more VMs and quickly deploy VMs with the CSR 1000v software pre-installed. This tool can speed the process of deploying Cisco CSR 1000v on multiple VMs.

COT provides a simple command-line interface to enter the VM attributes into the .ova file. COT can be run either in a LINUX shell or on Mac OS X. VMware ovftools must be installed.

Danger The Common OVF Tool (COT) is provided without official Cisco support. Use it at your own risk.

- Downloading COT, on page 89
- Editing the Basic Properties of Cisco CSR 1000v using COT, on page 89
- Editing the Custom Properties of Cisco CSR 1000v using COT, on page 90
- Deploying the Cisco CSR 1000v VM using COT, on page 93

COT Restrictions

- COT supports deployment of the OVA package directly onto an ESXi host. The tool does not support Citrix XenServer, KVM or Microsoft Hyper-V environments.

Downloading COT

Download and install the COT libraries and script according to the instructions on the GitHub site:

http://cot.readthedocs.io/en/latest/installation.html

Editing the Basic Properties of Cisco CSR 1000v using COT

Before deploying Cisco CSR 1000v using COT, you can edit the basic or custom properties of the Cisco CSR 1000v VM in the OVA package using COT.

To edit the basic properties of the OVA, use the **cot edit-properties** command.

cot edit-properties

-p *key1=value1*, **--properties** *key1=value1*

Sets properties using key value pairs. Example: **-p "login-username=cisco"** sets the login username using a key value pair.

-o *output*

Specifies the name or path to a new OVA package, if you are creating a new OVA instead of updating the existing OVA.

For more information on COT command **cot edit-properties**, see:
http://cot.readthedocs.io/en/latest/usage_edit_properties.html

Editing the Basic Properties of Cisco CSR 1000v using COT: Example

```
cot edit-properties csr1000v-universalk9.ova
-p "login-username=cisco"

-p "login-password=cisco"
-o csr1000v-universalk9-customized.ova
\# save modifications to a new OVA
cot info csr1000v-universalk9-customized.ova
```

```
  # verify the new values of properties in the OVA
(...)
Properties:
  <config-version>                                    "1.0"
  Router Name                                         ""
  Login Username                                      "cisco"
  Login Password                                      "cisco"
  Management Interface                                "GigabitEthernet1"
  Management VLAN                                     ""
  Management Interface IPv4 Address/Mask              ""
```

The table below shows the **cot edit-properties** command and arguments used in the above example.

Script Step	Description
`cot edit propertie s csr1000v-universalk9.ova`	Edits the basic environment properties of this OVA (csr1000v-universalk9.ova).
`-p "login-username=cisco"`	Sets the bootstrap login username.
`-p "login-password=cisco"`	Sets the bootstrap login password.
`-o "csr1000v-universalk9-customized.ova"`	Saves a modified OVA, which contains config commands from the text file.

Editing the Custom Properties of Cisco CSR 1000v using COT

Before doing the procedures shown in section Deploying the Cisco CSR 1000v VM using COT, on page 93, you can edit custom properties, for example to include Cisco IOS XE CLI commands.

To edit the custom properties of the OVA, use one of the following two commands:

- **cot edit-properties**; see cot edit-properties, on page 90.
- **cot inject-config**; see cot inject-config, on page 91.

cot edit-properties

Use the **cot edit-properties** command to pre-apply a small number of configuration commands to the OVA. (Otherwise, for a larger number of commands, consider using the **cot inject-config** command; see cot inject-config, on page 91.)

For further details about the **cot edit-properties** command, see http://cot.readthedocs.io/en/latest/usage_edit_properties.html .

Synopsis and Description

cot edit-properties *ova-filename*

-o *output*

Specifies the name or path to a new OVA package, if you are creating a new OVA instead of updating the existing OVA.

-c *config-file*

Specifies the name of a text file containing IOS XE commands to be added to the OVA.

Example

In this example, a previously created text file, iosxe_config.txt, containing IOS XE config commands is added to the OVA using the **cot edit-properties** command. Finally the **cot info** command is used to show the modified OVA.

```
$ cat iosxe_config.txt

interface GigabitEthernet1
no shutdown
ip address 192.168.100.10 255.255.255.0
!
ip route 0.0.0.0 0.0.0.0 GigabitEthernet1 192.168.100.1
$ cot edit-properties csr1000v-universalk9.ova \
     -o csr1000v-universalk9-customized.ova \
     -c iosxe_config.txt
$ cot info csr1000v-universalk9-customized.ova

...

Properties:
  <config-version>              "1.0"
  Router Name                   ""

...

  Intercloud Tunnel Interface Gateway IPv4 Address   ""
  <ios-config-0001>             "interface GigabitEthernet1"
  <ios-config-0002>             "no shutdown"
  <ios-config-0003>             "ip address 192.168.100.10 255.255.255.0"
  <ios-config-0004>             "ip route 0.0.0.0 0.0.0.0 GigabitEthernet1 192.168.100.1"
```

The table below shows the **cot edit properties** command and arguments used in the example.

Script Step	Description
`cot edit properties csr1000v-universalk9.ova`	Edits the custom environment properties of this OVA (csr1000v-universalk9.ova).
`-o "csr1000v-universalk9-customized.ova"`	New OVA, containing configuration commands from the text file.
`-c iosxe_config.txt`	Text file that contains IOS XE configuration commands. Each line of configuration in this file results in a entry such as com.cisco.csr1000v.ios-config-xxxx in the XML of the OVF.

cot inject-config

Use the **cot inject-config** command if you have a large set of configuration commands to pre-apply to the OVA; for example, if you want to add a complete running configuration. This is efficient in terms of file size and loading time as it uses plain text for the configuration commands (instead of XML). For further details about the **cot inject-config** command, see

http://cot.readthedocs.io/en/latest/usage_inject_config.html

Synopsis and Description

cot inject-config ova-filename

-o *output*

Specifies the name or path to a new OVA package, if you are creating a new OVA instead of updating the existing OVA.

-c *config-file*

Specifies the name of a text file, such as iosxe_config.txt , to be embedded in the OVA.

Example

In this example, the **cot inject-config** command adds Cisco IOS XE commands in text file iosxe_config.txt to the OVA.

```
$ cat iosxe_config.txt
interface GigabitEthernet1
no shutdown
ip address 192.168.100.10 255.255.255.0
!
ip route 0.0.0.0 0.0.0.0 GigabitEthernet1 192.168.100.1
$ cot inject-config csr1000v-universalk9.ova \

 -o csr1000v-universalk9-customized.ova \
      -c iosxe_config.txt
$ cot info csr1000v-universalk9-customized.ova
```

<.. other output snipped for brevity ..>

```
Files and Disks:                        File Size  Capacity Device
                                        --------- --------- --------------------
  csr1000v_harddisk.vmdk                 71.50 kB   8.00 GB harddisk @ SCSI 0:0
  bdeo.sh                                52.42 kB
  README-OVF.txt                          8.53 kB
  README-BDEO.txt                         6.75 kB
  cot.tgz                               116.78 kB
  csr1000v-universalk9.iso              484.80 MB           cdrom @ IDE 1:0
  config.iso                            350.00 kB           cdrom @ IDE 1:1
```

The table below shows the **cot inject-config** command and arguments used in the example.

Script Step	Description
`cot inject-config csr1000v-universalk9.ova`	Edits the custom environment properties of this OVA (csr1000v-universalk9.ova).
`-o "csr1000v-universalk9-customized.ova"`	Name of the new, modified OVA, containing config commands from the text file.
`-c iosxe_config.txt`	Name of the text file that contains IOS XE config commands.

Deploying the Cisco CSR 1000v VM using COT

To deploy the Cisco CSR 1000v VM, use the **cot deploy ... esxi** command as shown in the following step. Note that the following description provides general guidance. The exact steps that you need to perform may vary depending on the characteristics of your VMware environment and setup.

SUMMARY STEPS

1. Run the **cot deploy ... esxi** command to deploy the Cisco CSR 1000v. The script options are described at: http://cot.readthedocs.io/en/latest/usage_deploy_esxi.html

DETAILED STEPS

Procedure

Run the **cot deploy ... esxi** command to deploy the Cisco CSR 1000v. The script options are described at: http://cot.readthedocs.io/en/latest/usage_deploy_esxi.html

Also see the example below.

Note: The default values may vary depending on the Cisco CSR 1000v version.

Example

The table below shows an example **cot deploy** command, and its arguments, that is used to deploy a Cisco CSR 1000v VM in a vCenter environment.

Script Step	Description
`cot deploy`	
`-s '10.122.197.5/UCS/host/10.122.197.38'`	vCenter server 10.122.197.5, target host UCS/host/10.122.197.38
`-u administrator -p password`	Credentials for the ESXi server. If unspecified, COT will use your userid and prompt for a password.
`-n XE3.13`	Name of the newly created CSR VM.
`-c 1CPU-4GB`	OVF hardware config profile. If this is not specified, COT displays a list of available profiles and prompts you to select one.

`-N "GigabitEthernet1=VM Network"`
`-N "GigabitEthernet2=VM Network"`
`-N "GigabitEthernet3=VM Network"`
	Mapping each NIC in the Cisco CSR 1000v OVA to a vSwitch on the server.

`esxi`	Target hypervisor (currently always ESXi)

Script Step	Description
`~/Downloads/csr1000v-universalk9.ova`	OVA to deploy
`-ds=datastore38a`	Any ESXi-specific parameters—here, the datastore to use for disk storage.

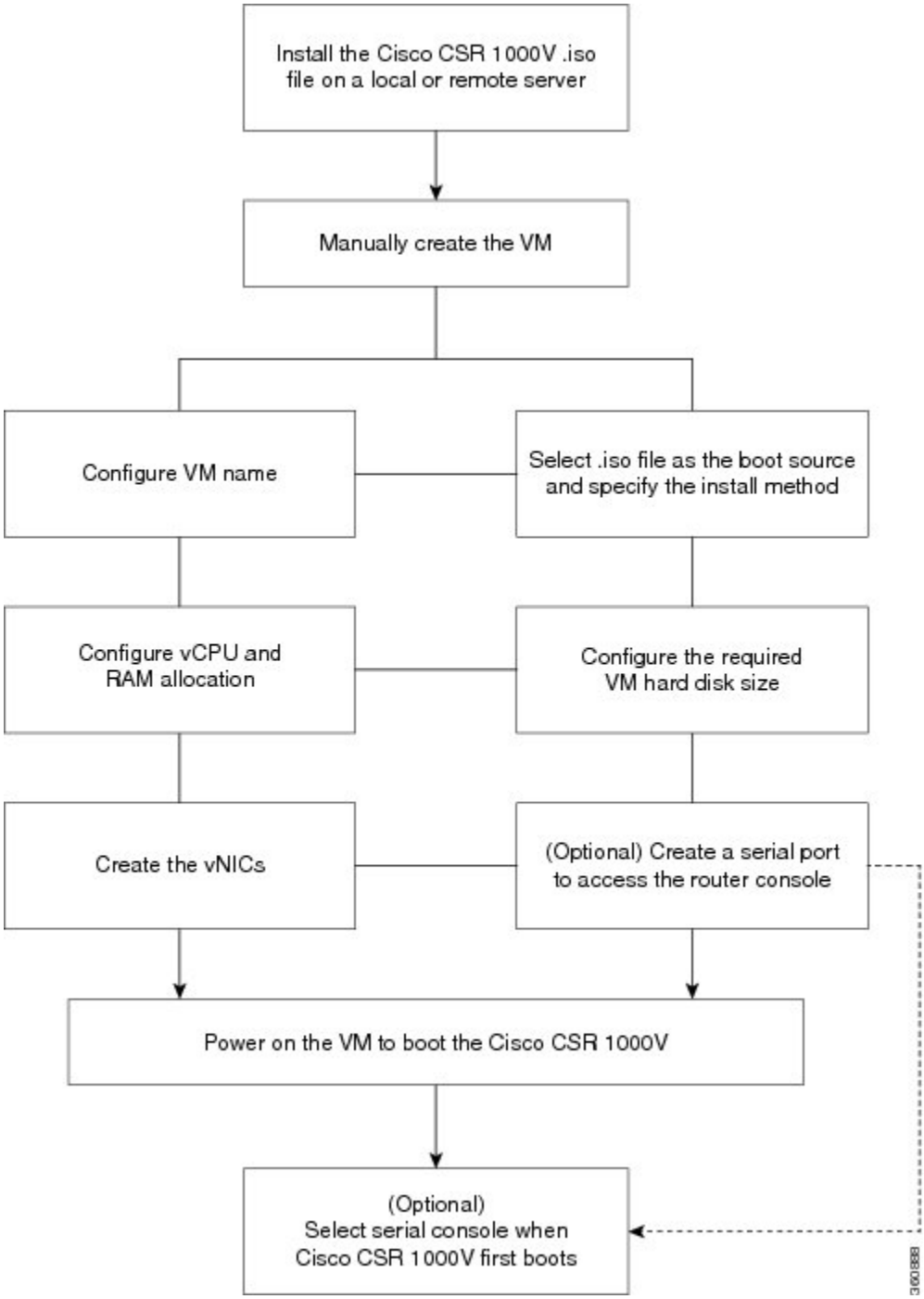
Manually Creating the VM and Installing the Cisco CSR 1000v Software Using the .iso File (VMware ESXi)

Overview of Tasks for Manually Creating the Cisco CSR 1000v VM

The figure below shows the typical high-level tasks required to manually create the Cisco CSR 1000v VM. The specific procedures, terminology and the order the steps are performed may differ depending on the hypervisor being used. See the sections following for detailed steps for creating the VM.

Note If you manually create the VM and you plan to use the Cisco CSR 1000v REST API, you must configure the HTTPS port using the Cisco IOS XE CLI.

Figure 5: Task Overview for Manually Creating the Cisco CSR 1000v VM



Manually Creating the Cisco CSR 1000v VM Using the .iso File (VMware ESXi)

The following steps are performed using VMware VSphere.

- Location: Store with the virtual machine

While the following procedure provides general guidance for how to deploy the Cisco CSR 1000v, the exact steps that you need to perform may vary depending on the characteristics of your VMware environment and setup. The steps and screen displays in this procedure are based on VMware ESXi 5.0.

SUMMARY STEPS

1. Download the CSR1000_esxi.iso file from the Cisco CSR 1000v software installation image package and copy it onto the VM Datastore.
2. In the VSphere client, select Create a New Virtual Machine option.
3. Under Configuration, select the option to create a Custom configuration, and click Next.
4. Under Name and Location, specify the name for the VM and click Next.
5. Under Storage, select the datastore to use for the VM. Click **Next**.
6. Under Virtual Machine Version, select Virtual Machine Version 8. Click **Next**.
7. Under Guest Operating System, select Linux and the "Other 2.6x Linux (64-bit) setting" from the drop-down menu. Click **Next**.
8. Under CPUs, select the following settings:
9. Under Memory, configure the supported memory size for your Cisco CSR 1000v release.
10. Under Network, allocate at least three virtual network interface cards (vNICs).
11. Under SCSI Controller, select LSI Logic Parallel. Click **Next**.
12. Under Select a Disk, click Create a new virtual disk.
13. Under Create a Disk, select the following:
14. Under Advanced Options, select SCSI (0:0) for the virtual device node.
15. On the Ready to Complete screen, click the Edit the virtual machine settings before completion. Click Continue checkbox.
16. In the Hardware tab, click **New CD/DVD Drive**.
17. In the Resources tab, click the CPU setting:
18. Click OK.
19. Click Finish.

DETAILED STEPS

Procedure

Step 1 Download the CSR1000_esxi.iso file from the Cisco CSR 1000v software installation image package and copy it onto the VM Datastore.

Step 2 In the VSphere client, select Create a New Virtual Machine option.

Step 3 Under Configuration, select the option to create a Custom configuration, and click Next.

Step 4 Under Name and Location, specify the name for the VM and click Next.

Step 5 Under Storage, select the datastore to use for the VM. Click **Next**.

Step 6 Under Virtual Machine Version, select Virtual Machine Version 8. Click **Next**.

Note
The Cisco CSR 1000v is not compatible with ESXi Server versions prior to 5.0.

Step 7 Under Guest Operating System, select Linux and the "Other 2.6x Linux (64-bit) setting" from the drop-down menu. Click **Next**.

Step 8 Under CPUs, select the following settings:

- Number of virtual sockets (virtual CPUs)
- Number of cores per socket

The number of cores per socket should always be set to 1, regardless of the number of virtual sockets selected. For example, a Cisco CSR 1000v with a 4 vCPU configuration should be configured as 4 sockets and 1 core per socket.

Click Next.

Step 9 Under Memory, configure the supported memory size for your Cisco CSR 1000v release.

Click **Next**.

Step 10 Under Network, allocate at least three virtual network interface cards (vNICs).

a) Select the number of vNICs that you want to connect from the drop-down menu.

Note
The VMware ESXi 5.0 interface only allows the creation of 4 vNICS during the initial VM creation. You can add more vNICs after the VM is created and the Cisco CSR 1000v is first booted.

b) Add the vNICs.

Select a different network for each vNIC.

Select the adapter type from the drop-down menu. See the requirements sections in this guide for the supported adapter type in your release.

c) Select all vNICs to connect at power-on.

d) Click **Next**.

Note
(Cisco IOS XE Release 3.10S and earlier) The first vNIC added is mapped to the GigabitiEthernet0 management interface on the Cisco CSR 1000v. All remaining vNICs are mapped to the Cisco CSR 1000v network interfaces when the VM is powered on and the router boots for the first time. For more information about how the vNICs on the VM map to the network interfaces on the router, see Mapping the Router Network Interfaces to vNICs, on page 251.

Note
You can add vNICs into the VM using vSphere while the Cisco CSR 1000v is running. For more information about adding vNICS to an existing VM, see the vSphere documentation.

Step 11 Under SCSI Controller, select LSI Logic Parallel. Click **Next**.

Step 12 Under Select a Disk, click Create a new virtual disk.

Step 13 Under Create a Disk, select the following:

- Capacity: Disk Size

  See the requirements sections in this guide for the virtual hard disk size required in your release.

- Disk Provisioning: select one of the following: Thick Provision Lazy Zeroed or Thick Provision Eager Zeroed.

  Note
  The Thin Provision option is not supported. The Thick Provision Eager Zeroed option takes longer to install but provides better performance.

- Location: Store with the Virtual Machine

Click Next.

Step 14 Under Advanced Options, select SCSI (0:0) for the virtual device node.

Step 15 On the Ready to Complete screen, click the Edit the virtual machine settings before completion. Click Continue checkbox.

Step 16 In the Hardware tab, click **New CD/DVD Drive**.

a) Select the Device Type that the VM will boot from:

Select the Datastore ISO file option to boot from the Cisco CSR 1000v .iso file. Browse to the location of the .iso file on the datastore set in step 1.

b) In the Device Status field, select the Connect at power on checkbox.

c) Select the Virtual Device Node CD/DVD drive on the host that the VM will boot from.

Step 17 In the Resources tab, click the CPU setting:

Set the Resource Allocation setting to Unlimited.

Step 18 Click OK.

Step 19 Click Finish.

The VM is now configured for the Cisco CSR 1000v and is ready to boot. The Cisco CSR 1000v is booted when the VM is powered on. See Booting the Cisco CSR 1000v and Accessing the Console, on page 143.

Note
To access and configure the Cisco CSR 1000v from the serial port on the ESXi host instead of the virtual VGA console, provision the VM to use this setting before powering on the VM and booting the router. For more information, see Booting the Cisco CSR 1000v and Accessing the Console, on page 143.

Increasing Performance on VMware ESXi Configurations

You can improve performance on VMware ESXi configurations by performing the following:

- Disable VMware ESXi power management.

  Choose the High Performance setting to disable power management in VMware ESXi 5.0, 5.1, 5.5, or 6.0. For more information, see the VMware Documentation .

VMware Requirements—Cisco IOS XE 3.x

The VMware requirements supported by Cisco CSR 1000v using old versions of Cisco IOS XE from 3.9 to 3.17 are shown in the following table:

Table 24: VMware Requirements for Cisco CSR 1000v (Cisco IOS XE versions 3.x)

Cisco CSR 1000v Release	VM Configuration Requirements
Cisco IOS XE Release 3.9S	VMware ESXi 5.0 8 GB virtual disk 4 virtual CPUs 4 GB of RAM 3 or more virtual network interface cards Single hard disk **Note** Multiple hard disk drives on a VM are not supported.
Cisco IOS XE Release 3.10S	VMware ESXi 5.0 or 5.1 8 GB virtual disk The following virtual CPU configurations are supported: • 1 virtual CPU, requiring 2.5 GB minimum of RAM • 4 virtual CPUs, requiring 4 GB minimum of RAM 3 or more virtual network interface cards Single hard disk **Note** Multiple hard disk drives on a VM are not supported.
Cisco IOS XE Release 3.11S	VMware ESXi 5.0 or 5.1 8 GB virtual disk The following virtual CPU configurations are supported: • 1 virtual CPU, requiring 2.5 GB minimum of RAM • 2 virtual CPUs, requiring 2.5 GB minimum of RAM • 4 virtual CPUs, requiring 4 GB minimum of RAM 3 or more virtual network interface cards Single hard disk **Note** Multiple hard disk drives on a VM are not supported.

Cisco CSR 1000v Release	VM Configuration Requirements
Cisco IOS XE Release 3.12 and 3.13	VMware ESXi 5.0, 5.1, or 5.5 8 GB virtual disk The following virtual CPU configurations are supported: • 1 virtual CPU, requiring 2.5 GB minimum of RAM • 2 virtual CPUs, requiring 2.5 GB minimum of RAM • 4 virtual CPUs, requiring 4 GB minimum of RAM • 8 virtual CPUs, requiring 4 GB minimum of RAM 3 or more virtual network interface cards Single hard disk **Note** Multiple hard disk drives on a VM are not supported.
Cisco IOS XE Release 3.14, 3.15, 3.16, 3.17	VMware ESXi 5.0, 5.1, 5.5 (VMware ESXi 5.5 update 3 is supported on Cisco IOS XE 3.16.1S and later, and on 3.17s and later.), 6.0 (VMware ESXi 6.0 is supported on Cisco IOS XE 3.16.1S and later, and 3.17S and later.) 8 GB virtual disk The following virtual CPU configurations are supported: • 1 virtual CPU, requiring 4 GB minimum of RAM • 2 virtual CPUs, requiring 4 GB minimum of RAM • 4 virtual CPUs, requiring 4 GB minimum of RAM • 8 virtual CPUs, requiring 4 GB minimum of RAM 3 or more virtual network interface cards Single hard disk **Note** Multiple hard disk drives on a VM are not supported.

VMware VM Requirements—Cisco IOS XE 3.x

The VMware tools supported by Cisco CSR 1000v using versions of Cisco IOS XE from 3.9 to 3.17 are shown in the table below.

Table 25: VMware Virtual Machine Requirements (Cisco IOS XE versions 3.x)

Cisco CSR 1000v Release	Supported Tools and Requirements	Supported vSwitch
Cisco IOS XE Release 3.9S	PC running the following: • VMware vSphere Client 5.0 Server running the following: • VMware ESXi 5.0 (For more information about server requirements, see the Cisco CSR 1000V Series Cloud Services Router Release Notes .) Installation Tool: • VMware vCenter	VMware standard switch VMware distributed switch
Cisco IOS XE Release 3.10S and 3.11S	PC running the following: • VMware vSphere Client 5.0 Server running the following: • VMware ESXi 5.0 or 5.1 Installation Tool: • VMware vCenter	VMware standard switch VMware distributed switch
Cisco IOS XE Release 3.12S through 3.17S Cisco IOS XE Denali 16.2	PC running the following: • VMware vSphere Client 5.0, 5.1, or 5.5 Server running the following:1 • VMware ESXi 5.0, 5.1, or 5.5 (VMware ESXi 5.5 update 3 is supported on Cisco IOS XE 3.16.1S and later, and on 3.17s and later), 6.0 (VMware ESXi 6.0 supported on Cisco IOS XE 3.16.1S and later, and 3.17S and later.) Installation Tool: • VMware vCenter	VMware standard switch VMware distributed switch

Installation Requirements—Cisco IOS XE 3.x

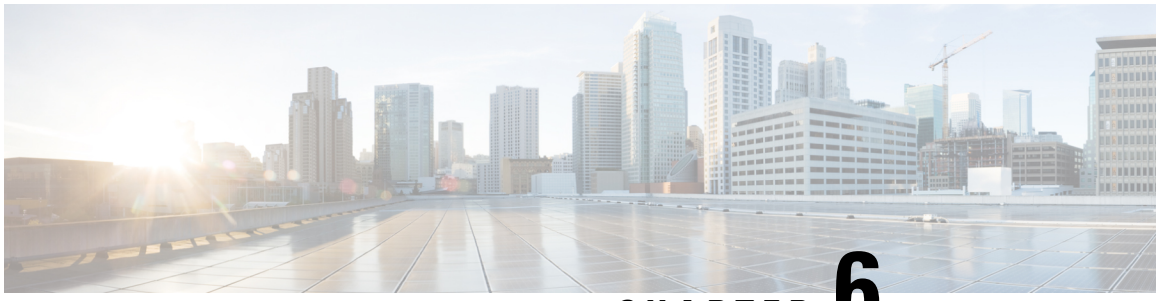
The table below lists the installation requirements for VMware ESXi using versions of Cisco IOS XE from 3.9 to 3.17. For Cisco IOS XE Denali 16.3 or later, see the respective release sections. For example, the *VMware Requirements - Cisco IOS XE Denali 16.3* section.

Note The Cisco CSR 1000v does not support Cisco IOS XE Denali 16.2.

Table 26: Installation Requirements for VMware ESXi (Cisco IOS XE 3.x)

VMware ESXi Requirement	Cisco IOS XE Release 3.9S	Cisco IOS XE Release 3.10S	Cisco IOS XE Release 3.11S	Cisco IOS XE Release 3.12S, 3.13S	Cisco IOS XE Release 3.14S, 3.15S, 3.16S, 3.17S
VMware ESXi version(s) supported	5.0	5.0, 5.1	5.0, 5.1	5.0, 5.1, 5.5	5.0, 5.1, 5.5 (VMware ESXi 5.5 update 3 is supported on Cisco IOS XE 3.16.1S and later, and on 3.17s and later.), 6.0 (VMware ESXi 6.0 supported on Cisco IOS XE 3.16.1S and later, and on 3.17S and later)
Supported vCPU configurations (The required vCPU configuration depends on the throughput license and technology package installed. For more information, see the data sheet for your release)	1 vCPU: requires minimum 4 GB RAM allocation	• 1 vCPU: requires minimum 2.5 GB RAM allocation (Not automatically supported when deploying the OVA) • 4 vCPUs: requires minimum 4 GB RAM allocation (If configuring Cisco Network Based Application Recognition (NBAR), or Cisco Application Visibility and Control (AVC), a 4-GB RAM allocation is required)	• 1 vCPU: requires minimum 2.5 GB RAM allocation (Not automatically supported when deploying the OVA) • 2 vCPUs: requires minimum 2.5 GB RAM allocation • 4 vCPUs: requires minimum 4 GB RAM allocation	• 1 vCPU: requires minimum 2.5 GB RAM allocation • 2 vCPUs: requires minimum 2.5 GB RAM allocation • 4 vCPUs: requires minimum 4 GB RAM allocation • 8 vCPUs: requires minimum 4 GB RAM allocation	• 1 vCPU: requires minimum 4 GB RAM allocation (ESXi 6.0 supported on Cisco IOS XE 3.16.1S and later) • 2 vCPUs: requires minimum 4 GB RAM allocation • 4 vCPUs: requires minimum 4 GB RAM allocation • 8 vCPUs: requires minimum 4 GB RAM allocation

VMware ESXi Requirement	Cisco IOS XE Release 3.9S	Cisco IOS XE Release 3.10S	Cisco IOS XE Release 3.11S	Cisco IOS XE Release 3.12S, 3.13S	Cisco IOS XE Release 3.14S, 3.15S, 3.16S, 3.17S
Virtual CPU cores required (Requires a 64-bit processor with Virtualization Technology (VT) enabled in the BIOS setup of the host machine.)	1	1	1	1	1
Virtual hard disk size	8 GB minimum	8 GB minimum	8 GB minimum	8 GB minimum	8 GB minimum
Supported vNICs	VMXNET3	VMXNET3	VMXNET3	VMXNET3	VMXNET3
Maximum number of vNICs supported	10	10	10	10	10
Default video, SCSI controller set	Required	Required	Required	Required	Required
Virtual CD/DVD drive installed	Required	Required	Required	Required	Required



CHAPTER 6

Installing the Cisco CSR 1000v in Citrix XenServer Environments

Citrix XenServer Support Information

Supported Releases

The Cisco CSR 1000v, using Cisco IOS XE 3.10S and later (Cisco IOS XE Denali 16.2 is not supported), is supported in the Citrix XenServer environment.

Other Support Information

The Cisco CSR 1000v installation on Citrix XenServer requires the manual creation of a VM and installation using the .iso file. Deploying the OVA template into a Citrix XenServer environment is not supported in this release.

The Cisco CSR 1000v supports the VIF vNIC type on the Citrix XenServer implementation.

The following Citrix XenServer features are supported:

- Virtual machine power-cycle
- Interface add and delete

Note This operation requires that the Cisco CSR 1000v is shutdown before performing interface add and delete.

- NIC bonding
- Virtual machine cloning

Only cold cloning is supported, meaning the VM must be powered down when the cloning takes place.

- Taking, restoring and deleting snapshots

Using Citrix XenServer, you can take a snapshot of the current state of the VM. Snapshots are supported when the Cisco CSR 1000v VM is either powered up or powered down.

- Remote storage
- Performance monitoring (CPU, network and disk)

Note The Cisco CSR 1000v does not support XenTools. The XenMotion operation is not supported on the Cisco CSR 1000v because it requires XenTools.

For more information, see Manually Creating the Cisco CSR 1000v VM Using the .iso File (Citrix XenServer), on page 107. For more information, see also the Citrix XenServer documentation.

Installation Requirements for Citrix XenServer

The following are the installation requirements for Citrix XenServer for Cisco CSR 1000V running on Cisco IOS XE 16.12 through 17.3.x images:

- Citrix XenServer version supported: Citrix XenServer 6.5 is recommended—tested and meets performance benchmarks. Citrix XenServer 6.2 is supported
- Supported vCPU configurations. (Also depends on the throughput license and technology package installed–see Datasheet). : 1 vCPU: requires minimum 4 GB RAM allocation; 2 vCPUs: requires minimum 4 GB RAM allocation; 4 vCPUs: requires minimum 4 GB RAM allocation
- Virtual CPU cores required: 1
- Supported vNICs : VIF-netfront (pmap)
- Maximum number of vNICs supported per VM instance: 7
- Virtual CD/DVD drive Installed: Required
- Virtual Disk—a 8 GB virtual disk is supported.
- Virtual CPU cores—1 vCPU is required

Note The Citrix XenServer requirements for early versions of Cisco IOS XE (before IOS XE Denali 16.3) are shown in Installation Requirements for Citrix XenServer: Cisco IOS XE 3.x, on page 108.

Manually Creating the Cisco CSR 1000v VM Using the .iso File (Citrix XenServer)

While the following procedure provides a general guideline for how to manually create the VM for the Cisco CSR 1000v, the exact steps that you need to perform may vary depending on the characteristics of your Citrix XenServer environment and setup. For more information, see the Citrix XenServer documentation.

To determine; for example, the number of vNICs, when installing the Cisco CSR 1000v on a Citrix XenServer VM , see the relevant "Installation Requirements for Citrix XenServer" section for your Cisco IOS XE Denali release.

Note The Cisco CSR 1000v does not support deploying the OVA file in KVM environments.

The following steps are performed using the Citrix XenCenter console.

SUMMARY STEPS

1. Download the .iso file from the Cisco CSR 1000v software installation image package and copy it onto a local or network device.
2. In the Citrix XenCenter console, to create a new VM, select the server, and click New VM.
3. Click Template. Scroll through the templates and select Other Install Media.
4. In the Name field, enter the name of the VM.
5. When prompted for the installation media, choose from one of the following:
6. Select the server where the VM will be placed.
7. Enter the number of vCPUs and memory settings. Click **Next**.
8. Add the virtual disks by inputting the following fields:
9. On the Networking screen, select the networks that will connect to the Cisco CSR 1000v through the vNICs.
10. Click **Finish**.

DETAILED STEPS

Procedure

Step 1 Download the .iso file from the Cisco CSR 1000v software installation image package and copy it onto a local or network device.

Step 2 In the Citrix XenCenter console, to create a new VM, select the server, and click New VM.

The Select a VM template screen displays.

Step 3 Click Template. Scroll through the templates and select Other Install Media.

Click **Next**.

Step 4 In the Name field, enter the name of the VM.

Step 5 When prompted for the installation media, choose from one of the following:

- Install from the ISO library or DVD drive
- Boot from network

Click **Next**.

Step 6 Select the server where the VM will be placed.

Select the checkbox for Place the VM on the server. Click Next.

Step 7 Enter the number of vCPUs and memory settings. Click **Next**.

Step 8 Add the virtual disks by inputting the following fields:

- Enter the description (optional).
- Select the virtual disk size from the pull-down menu.

  See the requirements sections in this guide for the supported number of vCPUs and memory requirements for your release.
- Enter the location of the virtual disk.

Click **Add** and then click **Next**.

Step 9 On the Networking screen, select the networks that will connect to the Cisco CSR 1000v through the vNICs.

See the requirements sections in this guide for the supported number of vCPUs and memory requirements for your release.

a) Select a network and click **Add Network**.
b) Select External and click **Next**.
c) Type in the network name. Click **Next**.
d) Select the NIC to use, the VLAN, and set the MTU value.

Step 10 Click **Finish**.

The new network is added. Repeat the procedure in the previous step for each vNIC.

For more information about booting the VM, see the documentation at: http://www.citrix.com/. When the VM is booted, the Cisco CSR 1000v begins the first-time boot process. See Booting the Cisco CSR 1000v and Accessing the Console, on page 143 to continue the boot process.

Installation Requirements for Citrix XenServer: Cisco IOS XE 3.x

Installation Requirements for Citrix XenServer (Cisco IOS XE 3.14, 3.15, 3.16, 3.17)

For installation procedures, see Manually Creating the Cisco CSR 1000v VM Using the .iso File (Citrix XenServer), on page 107.

- Citrix XenServer version supported: 6.2
- The following supported vCPU configurations also depend on the throughput license and technology package installed–see Datasheet.

  1 vCPU: requires minimum 4 GB RAM allocation

  2 vCPUs: requires minimum 4 GB RAM allocation

  4 vCPUs: requires minimum 4 GB RAM allocation
- Virtual CPU cores required: 1
- Virtual hard disk size: 8 GB minimum
- Supported vNICs : VIF
- Maximum number of vNICs supported per VM instance: 7
- Virtual CD/DVD drive Installed: Required

Installation Requirements for Citrix XenServer (Cisco IOS XE 3.13)

For installation procedures, see Manually Creating the Cisco CSR 1000v VM Using the .iso File (Citrix XenServer), on page 107.

- Citrix XenServer version supported: 6.2
- Supported vCPU configurations (also depends on the throughput license and technology package installed–see Datasheet) : 1 vCPU: requires minimum 2.5 GB RAM allocation

  2 vCPUs: requires minimum 2.5 GB RAM allocation

  4 vCPUs: requires minimum 4 GB RAM allocation
- Virtual CPU cores required: 1
- Virtual hard disk size: 8 GB minimum
- Supported vNICs : VIF
- Maximum number of vNICs supported per VM instance: 7
- Virtual CD/DVD drive Installed: Required

Installation Requirements for Citrix XenServer (Cisco IOS XE 3.11, 3.12)

For installation procedures, see Manually Creating the Cisco CSR 1000v VM Using the .iso File (Citrix XenServer), on page 107.

- Citrix XenServer version supported: 6.0.2, 6.1
- Supported vCPU configurations (also depends on the throughput license and technology package installed–see Datasheet) : 1 vCPU: requires minimum 2.5 GB RAM allocation

  2 vCPUs: requires minimum 2.5 GB RAM allocation

  4 vCPUs: requires minimum 4 GB RAM allocation
- Virtual CPU cores required: 1

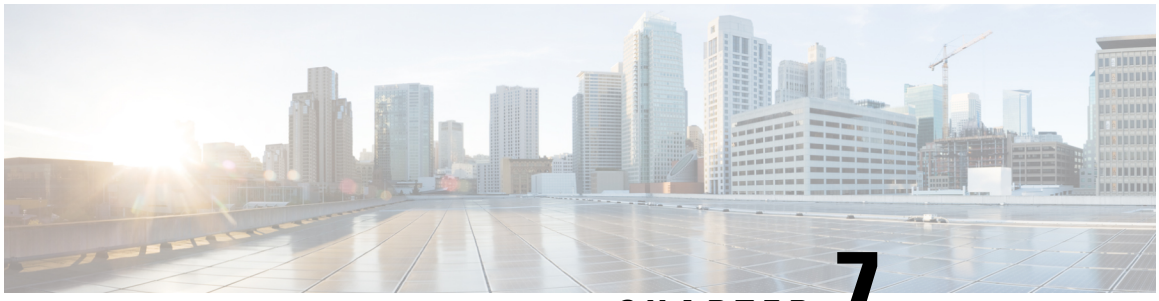
- Virtual hard disk size: 8 GB minimum
- Supported vNICs : VIF
- Maximum number of vNICs supported per VM instance: 7
- Virtual CD/DVD drive Installed: Required

Installation Requirements for Citrix XenServer (Cisco IOS XE 3.10)

For installation procedures, see Manually Creating the Cisco CSR 1000v VM Using the .iso File (Citrix XenServer), on page 107.

- Citrix XenServer version supported: 6.0.2
- Supported vCPU configurations. (Also depends on the throughput license and technology package installed–see Datasheet). : 4 vCPUs: requires 4 GB minimum RAM allocation
- Virtual CPU cores required: 1
- Virtual hard disk size: 8 GB minimum
- Supported vNICs : VIF
- Maximum number of vNICs supported per VM instance: 7
- Virtual CD/DVD drive Installed: Required

Note (Cisco IOS XE 3.10S Release and earlier) The network added to NIC0 maps to the Gigabit Ethernet 0 management interface on the Cisco CSR 1000v.



CHAPTER 7

Installing the Cisco CSR 1000v in KVM Environments

Kernel Virtual Machine Support Information

The CSR1000v supports the following Linux/KVM environments:

- Red Hat Enterprise Linux (RHEL)
- Red Hat Enterprise Virtualization (RHEV)
- Ubuntu (beginning with Cisco IOS XE Release 3.11S)

Red Hat Enterprise Linux (RHEL), an enterprise virtualization product produced by Red Hat, based on the Kernel-based Virtual Machine (KVM), is an open source, full virtualization solution for Linux on x86 hardware, containing virtualization extensions.

The Red Hat Enterprise Virtualization (RHEV) platform is a commercially packaged virtualization platform from Red Hat.

For more information on the KVM products and versions supported, see the "Installation Requirements for KVM" sections in the following few pages.

The Cisco CSR 1000v installation on KVM requires the manual creation of a VM and installation using the .iso file or the qcow2 file. Deploying the OVA template into a KVM environment is not supported.

The Cisco CSR 1000v supports the Virtio vNIC type on the KVM implementation. KVM supports a maximum of 26 vNICs.

KVM Support on OpenStack

(Cisco IOS XE 3.12S or later) The Cisco CSR 1000v supports the OpenStack environment. OpenStack support requires the .qcow2 installation file available on the Cisco.com download page. For more information, see Creating the Instance Using the OpenStack Command Line Tool , on page 121 and Creating the Instance Using the OpenStack Dashboard, on page 122.

Installation Requirements for KVM

The KVM requirements for Cisco CSR 1000V using Cisco IOS XE16.12 through 17.3 releases are as follows:

Note The KVM requirements for older versions of Cisco IOS XE (before IOS XE Denali 16.3) are shown in Installation Requirements for KVM—Cisco IOS XE 3.x, on page 131.

KVM Versions

Cisco IOS XE Release	KVM Version
Cisco IOS XE 17.3.x release	Linux KVM based on Red Hat Enterprise Linux 7.5 and 7.7 are recommended for release 17.3.1 - tested and meets performance benchmarks.
Cisco IOS XE 17.1.x and 17.2.x releases	Linux KVM based on Red Hat Enterprise Linux 7.4 is recommended for releases 17.1 and 17.2. These versions are tested and meets performance benchmarks.
Cisco IOS XE 16.12 release	Linux KVM based on Red Hat Enterprise Linux 7.4 is recommended - tested and meets performance benchmarks.

- vCPUs. The following vCPU configurations are supported:
  - 1 vCPU: requires minimum 4 GB RAM allocation
  - 2 vCPUs: requires minimum 4 GB RAM allocation
  - 4 vCPUs: requires minimum 4 GB RAM allocation
- Virtual CPU cores—1 vCPU is required
- Virtual hard disk size—8 GB minimum
- Supported vNICs—Virtio, ixgbe, ixgbevf or i40evf

Note If a vNIC with an i40evf driver is used, the maximum number of physical VLANs is limited to 512, shared across all VFs, and the number of VLANs for a VF can be further limited by the host (PF) driver for untrusted VFs. The latest Intel i40e PF driver limits untrusted VFs to a maxium of 8 VLANs/sub-interfaces.

- Maximum number of vNICs supported per VM instance—26
- Virtual CD/DVD drive installed (applicable only when installing using an .iso file)—required

Creating a Cisco CSR 1000v KVM Instance

Creating the Cisco CSR 1000v VM Using the Self-installing .Run Package

The Cisco CSR 1000v KVM Installer package (with **.run** extension) is a self-installing CSR package for KVM.

Default or Interactive

Installing a CSR instance using the **.run** package provides two options:

- Default mode

  Installation uses the bundled CSR image file and one of the default VM configuration options (small, medium, large, xlarge) described in the procedure below.

- Interactive mode

  Allows customization of VM configuration and option to install the bundled CSR image file or a separate .qcow2 image.

Installation Procedure

The following steps are performed on the KVM server.

Prerequisites

Download the .run executable from the Cisco CSR 1000v software installation image package and copy it onto a local device.

SUMMARY STEPS

1. Run the .run executable to launch the CSR VM.

DETAILED STEPS

Procedure

Run the .run executable to launch the CSR VM.

Example:

```
csr1000v-universalk9.03.16.01a.S.155-3.S1a-ext.run <option>
```

Values for <option>:

- **interactive**—Interactive mode
- **small**—Deploy CSR with: 1 vCPU, 4 GB RAM, 3 vNICs
- **medium**—Deploy CSR with: 2 vCPU, 4 GB RAM, 3 vNICs
- **large**—Deploy CSR with: 4 vCPU, 4 GB RAM, 3 vNICs
- **xlarge**—Deploy CSR with: 4 vCPU, 8 GB RAM, 3 vNICs

Example of Interactive Mode Installation

Example:

```
./csr1000v-universalk9.03.16.01a.S.155-3.S1a-ext.run interactive
Verifying archive integrity...  100%   All good.
Uncompressing KVM Installer Package  100%
Setting up installation of CSR on KVM
Tuning Daemon is already running
A tuning daemon process supports the automation of CSR VM vCPU and vhost vCPU affinity setting on
Ubuntu host OS and RHEL.
Please enter the appropriate values for the options below:
Enter the VM name [csr_29405]: my_csr_vm
Enter the image location
[/var/lib/libvirt/images/csr1000v-universalk9.03.16.01a.S.155-3.S1a-ext.qcow2_csr_29405]:
Enter the number of vCPUs [1]: 4
Enter the value for RAM [4096]: 8192
Enter number of vnics [3]: 2
Options for virtual bridge:
virbr0
vnet1
Enter bridge for vnic 1 [virbr0]:
Options for virtual bridge:
virbr0
vnet1
Enter bridge for vnic 2 [virbr0]:
Options for virtual bridge:
virbr0
vnet1
Creating VM my_csr_vm
Do you want to start the CSR?[y]
Starting CSR my_csr_vm...
VM my_csr_vm started
```

Creating the Cisco CSR 1000v VM Using the virt-manager GUI Tool

Creating the Cisco CSR 1000v VM Using virt-manager with qcow2 or ISO Image

Before you begin

Download and install the virt-manager RPM package on the KVM server.

Download the .qcow2 or .iso image from the Cisco CSR 1000v software installation image package and copy it onto a local or network device.

Procedure

Step 1 Launch the virt-manager GUI. Click **Create a new virtual machine**.

Step 2 Do one of the following: (For **.qcow2**) Select **Import existing disk image**. (For **.iso**) Select **Local install media (ISO image or CDROM)**.

Step 3 Select the CSR qcow2 or iso file location.

Step 4 Configure the memory and CPU parameters.

Step 5 Configure virtual machine storage.

Step 6 Click **Finish**.

Note
To add additional hardware before creating the VM, select **Customize configuration before install** before clicking **Finish**. If this option is selected, then the next screen displays an **Add Hardware** button that can be used one or more times to add various hardware options, such as additional disks or a serial port interface (see the following sections).

Step 7 Access the Cisco CSR 1000v console by using one of the following:

a) (If using the virtual console) Double-click the VM instance to access the VM console

b) (If using the serial console) See Booting the Cisco CSR 1000v and Accessing the Console, on page 143

What to do next

Creating the Cisco CSR 1000v VM Using virt-manager—Add Serial Port

Enables access to the CSR by adding a serial console. See Booting the Cisco CSR 1000v as the VM, on page 143.

SUMMARY STEPS

1. Click **Add Hardware**.
2. Select the **Serial** option from the menu.
3. From the **Device type** drop-down menu, select **TCP net console (tcp)**.
4. Specify the port number, and select the **Use Telnet** checkbox.
5. Click **Finish**.
6. After adding all necessary hardware, click **Begin Installation**.

DETAILED STEPS

Procedure

Step 1 Click **Add Hardware**.
Step 2 Select the **Serial** option from the menu.
Step 3 From the **Device type** drop-down menu, select **TCP net console (tcp)**.
Step 4 Specify the port number, and select the **Use Telnet** checkbox.
Step 5 Click **Finish**.
Step 6 After adding all necessary hardware, click **Begin Installation**.

Creating the Cisco CSR 1000v VM Using virt-manager--Add Hard Disk

Describes the optional steps after selecting **Customize configuration before install** .

Before you begin

- Perform the task Creating a Bootstrap Day0 Configuration for virt-manager, on page 117 —Using .qcow2 or .iso Image, but during the task steps before clicking "Finish", select the **Customize configuration before install** option . A screen with "Add Hardware" button appears.

SUMMARY STEPS

1. Click **Add Hardware**.
2. Select the **Storage** option from the menu.
3. Select **Select managed or other existing storage** checkbox.
4. (Applicable only when adding a Bootstrap Day0 configuration) Click the **Browse** button and navigate to the **csr_config.iso** location. From the **Device type** drop-down menu, select the **IDE CDROM** option.
5. Click **Finish**.
6. After adding all necessary hardware, click **Begin Installation**.

DETAILED STEPS

Procedure

Step 1 Click **Add Hardware**.
Step 2 Select the **Storage** option from the menu.
Step 3 Select **Select managed or other existing storage** checkbox.
Step 4 (Applicable only when adding a Bootstrap Day0 configuration) Click the **Browse** button and navigate to the **csr_config.iso** location. From the **Device type** drop-down menu, select the **IDE CDROM** option.
Step 5 Click **Finish**.

Step 6 After adding all necessary hardware, click **Begin Installation**.

Creating a Bootstrap Day0 Configuration for virt-manager

This procedure provides additional steps to be executed within Creating the Cisco CSR 1000v VM Using virt-manager with qcow2 or ISO Image, on page 115.

The following steps are performed on the KVM server.

SUMMARY STEPS

1. Create **iosxe_config.txt** or **ovf-env.xml** file. (For details, see Bootstrap Properties, on page 123.)
2. Create a disk image from this file using below command:
3. (This step must be performed within Creating the Cisco CSR 1000v VM Using virt-manager with qcow2 or ISO Image, on page 115 .)

DETAILED STEPS

Procedure

Step 1 Create **iosxe_config.txt** or **ovf-env.xml** file. (For details, see Bootstrap Properties, on page 123.)

Step 2 Create a disk image from this file using below command:

Example:

```
mkisofs -l -o /my/path/csr_config.iso <configuration_filename>
```

Step 3 (This step must be performed within Creating the Cisco CSR 1000v VM Using virt-manager with qcow2 or ISO Image, on page 115 .)

Mount the **csr_config.iso** as an additional disk during creation of the CSR virtual machine.

Creating the Cisco CSR 1000v VM Using virt-install with qcow2 Image

- Download and install the virt-install RPM package on the KVM server.
- Download the .**qcow2** image from the Cisco CSR 1000v software installation image package and copy it onto a local or network device.

Before you begin

Procedure

Using the virt-install command, create the instance and boot, using the following syntax:

Example:

```
virt-install                    \
     --connect=qemu:///system   \
     --name=my_csr_vm           \
     --os-type=linux            \
     --os-variant=rhel4         \
     --arch=x86_64              \
     --cpu host                 \
     --vcpus=1,sockets=1,cores=1,threads=1   \
     --hvm                      \
     --ram=4096                 \
     --import                   \
     --disk path=<path_to_csr1000v_qcow2>,bus=ide,format=qcow2   \
     --network bridge=virbr0,model=virtio                        \
     --noreboot
```

(Optional) To configure a Bootstrap Day0 configuration, perform the steps described in Creating a Bootstrap Day0 Configuration for virt-install, on page 119.

After the installation is complete, the CSR VM will be shutdown. You can start the CSR VM using the **virsh start** command.

What to do next

Red Hat Enterprise Linux—Setting Host Mode

Due to an issue specific to Red Hat Enterprise Linux, when launching the Cisco CSR1000v in a Red Hat Enterprise Linux environment using **virt-install**, set the host mode as follows:

- In Red Hat Enterprise Linux 6, use:

```
--cpu host
```

- In Red Hat Enterprise Linux 7, use:

```
--cpu host-model
```

Creating the Cisco CSR 1000v VM Using virt-install with ISO Image

Before you begin

- Download and install the virt-install RPM package on the KVM server.
- Download the **.iso** image from the Cisco CSR 1000v software installation image package and copy it onto a local or network device.

Procedure

Step 1 Create a 8G disk image in **.qcow2** format using the **qemu-img** command.

Example:

```
qemu-img create -f qcow2 csr_disk.qcow2 8G
```

Step 2 Use the **virt-install** command to install the CSR. This requires the correct permissions to create a new VM. The following example creates a 1 vCPU CSR with 4G of RAM, one network interface, and one serial port.

Example:

```
virt-install                \
 --connect=qemu:///system   \
 --name=my_csr_vm           \
 --description "Test VM"    \
 --os-type=linux            \
 --os-variant=rhel4         \
 --arch=x86_64              \
 --cpu host                 \
 --vcpus=1,sockets=1,cores=1,threads=1    \
 --hvm                      \
 --ram=4096                 \
 --cdrom=<path_to_csr1000v_iso>           \
 --disk path=csr_disk.qcow2,bus=virtio,size=8,sparse=false,cache=none,format=qcow2    \
 --network bridge=virbr0,model=virtio     \
 --noreboot
```

(Optional) To configure a Bootstrap Day0 configuration, perform the steps described in Creating a Bootstrap Day0 Configuration for virt-install, on page 119.

The **virt-install** command creates a new VM instance and the CSR installs the image onto the specified disk file. After the installation is complete, the CSR VM will be shutdown. You can start the CSR VM using the **virsh start** command.

Step 3

What to do next

Red Hat Enterprise Linux—Setting Host Mode

Due to an issue specific to Red Hat Enterprise Linux, when launching the Cisco CSR1000v in a Red Hat Enterprise Linux environment using **virt-install**, set the host mode as follows:

- In Red Hat Enterprise Linux 6, use:

```
--cpu host
```

- In Red Hat Enterprise Linux 7, use:

```
--cpu host-model
```

Creating a Bootstrap Day0 Configuration for virt-install

This procedure provides additional steps to execute within one of the following procedures, as noted within the procedures:

- Creating the Cisco CSR 1000v VM Using virt-install with qcow2 Image, on page 117
- Creating the Cisco CSR 1000v VM Using virt-install with ISO Image, on page 118

Note The CSR1000v will read the day 0 configuration during initial bootup and will save the configuration after bootup.

In order to bootstrap, perform the following steps on the KVM server.

SUMMARY STEPS

1. Create an **iosxe_config.txt** or **ovf-env.xml** file. (For details, see Bootstrap Properties, on page 123.)
2. Create a disk image from the file using following command:
3. (This step must be performed within the VM creation procedure (see the options indicated above).

DETAILED STEPS

Procedure

Step 1 Create an **iosxe_config.txt** or **ovf-env.xml** file. (For details, see Bootstrap Properties, on page 123.)

Step 2 Create a disk image from the file using following command:

Example:

```
mkisofs -l -o /my/path/csr_config.iso <configuration_filename>
```

Step 3 (This step must be performed within the VM creation procedure (see the options indicated above).

Add an additional disk parameter to the **virt-install** command to include the csr_config.iso disk image, as follows:

Example:

```
--disk path=/my/path/csr_config.iso,device=cdrom,bus=ide
```

Creating the Cisco CSR 1000v on OpenStack

Selecting a Cisco CSR 1000v Installation Image

There are two different installation image packages available for downloading. Each package contains a different qcow2 file.

- `csr1000v-universalk9.16.03.01a.qcow2`

  —choose an image such as this (for Cisco IOS XE Denali 16.3.1) to use a virtual console. This is recommended for later use with the OpenStack dashboard.

- `csr1000v-universalk9.16.03.01a-serial.qcow2`

  —choose an image such as this (for Cisco IOS XE Denali 16.3.1) to use a serial console to access the VM. This is useful in the lab or if you are using the Cisco Modeling Tool.

Creating the Instance Using the OpenStack Command Line Tool

Although the following procedure provides a general guideline for how to create the Cisco CSR 1000v tenant instance, the exact steps that you need to perform may vary depending on the characteristics of your KVM environment and setup. For more information, see the OpenStack documentation. See "Installation Requirements" sections.

The following steps are performed using the Nova (OpenStack Compute) console on your server.

Procedure

Step 1 Download the .qcow2 file from the Cisco CSR 1000v software installation image package and copy it onto a local or network device

Step 2 Create the Nova flavor using the **nova flavor-create**command

nova flavor-create <*flavor_name*> <*flavor_id*> <*ram size MB*> <*disk size GB*> <*num_ vCPUs*>

The disk size should be set to 0 for the Cisco CSR 1000V to boot. The following command example creates a KVM instance with 4096 MB RAM, a disk size of 0 and 2 vCPUs configured:

nova flavor-create csr_flavor 6 4096 0 2

Step 3 Enter the **nova flavor-list** command to verify that the nova flavor created the previous step is available

Step 4 Use the **glance** command, to create the OpenStack image

glance image-create --name <*image_name*> **--disk-format qcow2 --container-format bare --file** <*Location-of-img-file*>

The following example creates an OpenStack image using the Cisco CSR 1000v installation file:

Example:

```
glance image-create --namecsr_image --disk-format qcow2 --container-format bare
  --file /opt/stack/csr/files/images/csr1000v-universalk9.03.12.00.S.154-2.S-std.qcow2
```

Step 5 Use the **nova boot** command, to create the instance and boot

nova boot <*instance_name*> **--image** <*image_id*> **--flavor** <*flavor_id*> **--nic net-id=**<*uuid*> **--config-drive=**<**true**/**false**> **--file**<*configuration_file_name*>

The **--config-drive** option can be used to specify that the configuration is loaded on the Cisco CSR 1000v when it comes up. Set the **--config-drive** option to “true” and specify the name of the configuration file in which you enter the router configuration to be booted. There are two possible formats for the configuration file:

- “ovf-env.xml” (OVF format)
- “iosxe_config.txt”

Note
For details , see Bootstrap Properties, on page 123 and subsequent sections.

Note
These file names are hard-coded and required for the config-drive settings to boot.

Prior to Cisco IOS XE 3.16S, you could specify only one of the two configuration files in the **nova boot** command. Beginning with Cisco IOS XE 3.16S, and including Cisco IOS XE Denali 16.3.1 and later, you can specify both configuration files in the **nova boot** command line—for example:

Example:

```
nova boot csr-vm-316 --image csr-316 --flavor csr.2vcpu.4gb
  --nic port-id=6773be11-7b95-48cd-b372-fb8a3cae2b50 --config-drive=true
  --file ovf-env.xml=/home/stack/conf_files/ut/ovf-env.xml
   --file iosxe_config.txt=/home/stack/conf_files/ut/iosxe_config.txt
```

Example:

This example shows the booting of the Cisco CSR 1000v image on OpenStack with the “ovf-env.xml” file containing the router configuration:

```
nova boot csr_instance --image csr_image --flavor 6 --nic net-id=546af738-bc0f-43cf-89f2-1e2c747d1764

 --config-drive=true --file ovf-env.xml=/opt/stack/csr/files/ovf-env.xml
```

Example:

The following example boots the Cisco CSR 1000v image on OpenStack with the “iosxe_config.txt” file containing the router configuration:

```
nova boot csr_instance --image csr_image --flavor 6 --nic net-id=546af738-bc0f-43cf-89f2-1e2c747d1764

 --config-drive=true --file iosxe_config.txt=/opt/stack/iosxe_config.txt
```

The Cisco CSR 1000v begins the boot process. See Booting the Cisco CSR 1000v as the VM, on page 143.

After the OpenStack image is created, you can access the instance on your OpenStack dashboard.

Creating the Instance Using the OpenStack Dashboard

Perform the following steps to create the instance using the OpenStack dashboard.

Note To configure the KVM to run with config-drive, you must select the serial image and use the procedure described in the Creating the Instance Using the OpenStack Command Line Tool , on page 121.

Procedure

Step 1 Download the .qcow2 file from the Cisco CSR 1000v software installation image package and copy it onto a local or network device.

Step 2 From the OpenStack dashboard, access the OpenStack console.

Step 3 Login as the admin onto the OpenStack console.

Step 4 Create a new flavor using the **Flavor Create** tab on the screen, and specify the *<flavor_name> <flavor_id> <ram size MB> <disk size GB> <num_ vCPUs>*.

See the "Installation Requirements" section required for your version of IOS XE Installing the Cisco CSR 1000v in KVM Environments, on page 111. The disk size should be set to 0 for the Cisco CSR 1000v to boot. as in the tables 6-1 and 6-2.

Select the required flavor from the **System Panel** > **Flavors** tab.

Step 5 Create a new image using the **Image Create** tab on the screen.

Specify the location of the image, the disk format (qcow2) and container-format (raw).

Select the **System Panel** > **Images** tab. The image should show up on the list of images shown on the screen.

Step 6 Create a new instance using the **Instance Create** tab on the screen.

Specify the image, the flavor, and the appropriate network interfaces to be attached to the instance.

Select the **System Panel** > **Instances** tab. The instance should show up on the list of instances shown on the screen, and you should be able to access the console by clicking on the instance name.

Step 7 To launch the instance, select the instance and select **Launch Instance**.

Click the **Details** tab. Review the instance information to ensure it is correct. When you ready to launch the instance, click **Launch**.

The instance is launched and the Cisco CSR 1000v begins the boot process. See Booting the Cisco CSR 1000v as the VM, on page 143.

Troubleshooting for Creating the Instance using OpenStack

The following issues may occur when creating the instance using the OpenStack Dashboard (see Creating the Instance Using the OpenStack Dashboard, on page 122).

- If you do not see any output on the OpenStack dashboard's console it may be due to you having previously selected the incorrect type of image. Select the virtual qcow2 image (not the serial qcow2 image) before following the steps in Creating the Instance Using the OpenStack Dashboard, on page 122.

Bootstrapping the CSR Configuration

Bootstrap Properties

The Cisco CSR 1000v bootstrap properties are specified in the **ovf-env.xml** file. For an example **ovf-env.xml** file, see the *Example ovf-env.xml File* section.

Table 27: Bootstrap Properties

Property	Description
console	(Cisco IOS XE 3.17S and later) Configures the console mode. Possible values: auto, virtual, serial
domain-name	Domain name of the router.
enable-scp-server	Enables the IOS SCP feature.

Property	Description
enable-ssh-server	Enables remote login using SSH and disables remote login via Telnet. Requires that the login username and password are set.
hostname	Hostname of the router.

ios-config	Enables execution of a Cisco IOS command.
	To execute multiple commands, use multiple instances of ios-config, with a number appended to each instance—for example, ios-config-1, ios-config-2. The commands are executed in numerical order according to the appended number.
	Example
	`ios-config-1="username cisco priv 15 pass ciscoxyz"`
	`ios-config-2="ip scp server enable"`
	`ios-config-3="ip domain lookup"`
	`ios-config-4="ip domain name cisco.com”`

license	(Cisco IOS XE 3.13S and later) Configures the license technology level that is available when the Cisco CSR 1000v boots.
login-password	Login password for the router.
login-username	Login username for the router.
mgmt-interface	Designates the management interface for the Cisco CSR 1000v. The format must be GigabitEthernetx or GigabitEthernetx.xxx. **Note** Beginning with Cisco IOS XE 3.11S, the GigabitEthernet0 interface is no longer supported.
mgmt-ipv4-addr	Management gateway address/mask in IPv4 format for the GigabitEthernet0 management interface.
mgmt-ipv4-gateway	IPv4 management default gateway address. If using DHCP, enter “dhcp” in the field.
mgmt-ipv4-network	Configures the IPv4 Network (such as “192.168.2.0/24” or “192.168.2.0 255.255.255.0”) that the management gateway should route to. If not specified, the default route (0.0.0.0/0) is used.
mgmt-vlan	Configures the dot1Q VLAN interface. Requires the management interface to be configured using the GigabitEthernetx.xxx format.
pnsc-agent-local-port	(Optional) Configures the Cisco Prime Network Services Controller service agent SSL port on the local Cisco CSR 1000v to receive policies from the service manager. This setting is used if you plan to remotely manage the Cisco CSR 1000v using the Cisco Prime Network Services Controller.

Property	Description
pnsc-ipv4-addr	Configures the IP address of the Cisco Prime Network Services Controller. This setting is used if you plan to remotely manage the Cisco CSR 1000v using the Cisco Prime Network Services Controller.
pnsc-shared-secret-key	Configures the Cisco Prime Network Services Controller shared secret key for the Cisco Prime Network Services Controller agent to set the SSL certificate from the controller. This setting is used if you plan to remotely manage the Cisco CSR 1000v using the Cisco Prime Network Services Controller.
privilege-password	Configures the password for privileged (enable) access.
remote-mgmt-ipv4-addr	(Optional) Configures the IP address used for remote management of the Cisco CSR 1000v by the REST API or by the Cisco Prime Network Services Controller. The address must be in the same subnet as the management interface address. **Note** Beginning with Cisco IOS XE 3.13S, this option is not used if configuring the shared management interface to support REST API.
resource-template	(Cisco 3.16S2 and later) Configures the Resource Template. Possible values: default, service_plane_medium, service_plane_heavy

Example ovf-env.xml File

```
<?xml version="1.0" encoding="UTF-8"?>
<Environment
   xmlns:oe="http://schemas.dmtf.org/ovf/environment/1">
    <PropertySection>
        <Property oe:key="com.cisco.csr1000v.license.1" oe:value="security"/>
        <Property oe:key="com.cisco.csr1000v.console.1" oe:value="serial"/>

<Property oe:key="com.cisco.csr1000v.config-version.1" oe:value="1.0"/>
        <Property oe:key="com.cisco.csr1000v.domain-name.1" oe:value=""/>
        <Property oe:key="com.cisco.csr1000v.enable-scp-server.1" oe:value="False"/>
        <Property oe:key="com.cisco.csr1000v.enable-ssh-server.1" oe:value="False"/>
        <Property oe:key="com.cisco.csr1000v.hostname.1" oe:value="lab"/>
        <Property oe:key="com.cisco.csr1000v.license.1" oe:value="ax"/>
        <Property oe:key="com.cisco.csr1000v.login-password.1" oe:value=""/>
        <Property oe:key="com.cisco.csr1000v.login-username.1" oe:value="lab"/>
       <Property oe:key="com.cisco.csr1000v.mgmt-interface.1" oe:value="GigabitEthernet1"/>

       <Property oe:key="com.cisco.csr1000v.mgmt-ipv4-addr.1" oe:value="172.25.223.251/25"/>

       <Property oe:key="com.cisco.csr1000v.mgmt-ipv4-gateway.1" oe:value="172.25.223.129"/>

        <Property oe:key="com.cisco.csr1000v.mgmt-ipv4-network.1" oe:value=""/>
        <Property oe:key="com.cisco.csr1000v.mgmt-vlan.1" oe:value=""/>
        <Property oe:key="com.cisco.csr1000v.pnsc-agent-local-port.1" oe:value=""/>
        <Property oe:key="com.cisco.csr1000v.pnsc-ipv4-addr.1" oe:value=""/>
        <Property oe:key="com.cisco.csr1000v.pnsc-shared-secret-key.1" oe:value=""/>
```

```
        <Property oe:key="com.cisco.csr1000v.privilege-password.1" oe:value=""/>
        <Property oe:key="com.cisco.csr1000v.remote-mgmt-ipv4-addr.1" oe:value=""/>
        <Property oe:key="com.cisco.csr1000v.resource-template.1"
oe:value="service_plane_medium"/>
        <Property oe:key="com.cisco.csr1000v.ios-config-0001" oe:value="logging buffered
10000"/>
       <Property oe:key="com.cisco.csr1000v.ios-config-0002" oe:value="hostname uut-ovf"/>

        <Property oe:key="com.cisco.csr1000v.ios-config-0003" oe:value="ip domain-name
cisco.com"/>
       <Property oe:key="com.cisco.csr1000v.ios-config-0004" oe:value="crypto key generate
 rsa modulus 1024"/>
        <Property oe:key="com.cisco.csr1000v.ios-config-0005" oe:value="interface
GigabitEthernet2"/>
       <Property oe:key="com.cisco.csr1000v.ios-config-0006" oe:value="ip address 10.0.0.5
 255.255.255.0"/>
        <Property oe:key="com.cisco.csr1000v.ios-config-0007" oe:value="no shut"/>
        <Property oe:key="com.cisco.csr1000v.ios-config-0008" oe:value="exit"/>
        <Property oe:key="com.cisco.csr1000v.ios-config-0009" oe:value="ip route 0.0.0.0
0.0.0.0 10.0.0.1"/>
    </PropertySection>
</Environment>
```

Example iosxe_config.txt File

```
hostname ultra-ios_cfg
license smart enable
username lab privilege 15 password lab
ip domain-name cisco.com
crypto key generate rsa modulus 1024
interface GigabitEthernet1
ip address 10.0.0.5 255.255.255.0
no shut
exit
ip route 0.0.0.0 0.0.0.0 10.0.0.1
line vty 0 4
 login local
exit
```

Increasing the KVM Configuration Performance

You can increase the performance for a Cisco CSR 1000v running in a KVM environment by changing some settings on the KVM host. These settings are independent of the IOS XE configuration settings on the CSR 1000v instance.

Note In Cisco IOS XE Release 3.13S and earlier, the CSR 1000v instance does not support jumbo packets larger than 1518 bytes for KVM on a Virtio interface. The packets larger than 1518 bytes are dropped.

To improve the KVM configuration performance, Cisco recommends that you:

- Enable vCPU pinning
- Enable emulator pinning
- Enable numa tuning. Ensure that all the vCPUs are pinned to the physical cores on the same socket.

- Set hugepage memory backing
- Use virtio instead of IDE
- Use graphics VNC instead of SPICE
- Remove unused devices USB, tablet etc.
- Disable memballoon

Note These settings might impact the number of VMs that you can be instantiate on a server.

Tuning steps are most impactful for a small number of VMs that you instantiate on a host.

In addition to the above mentioned, do the following:

Enable CPU Pinning

Increase the performance for the KVM environments by using the KVM CPU Affinity option to assign a virtual machine to a specific processor. To use this option, configure CPU pinning on the KVM host.

In the KVM host environment, use the following commands:

- **virsh nodeinfo**: To verify the host topology to find out how many vCPUs are available for pinning by using the following command.
- **virsh capabilities**: To verify the available vCPU numbers.
- **virsh vcpupin** <*vmname*> <*vcpu#*> <*host core#*>: To pin the virtual CPUs to sets of processor cores.

  This KVM command must be executed for each vCPU on your Cisco CSR 1000v. The following example pins virtual CPU 1 to host core 3:

  virsh vcpupin csr1000v 1 3

  The following example shows the KVM commands needed if you have a Cisco CSR 1000v configuration with four vCPUs and the host has eight cores:

  virsh vcpupin csr1000v 0 2

  virsh vcpupin csr1000v 1 3

  virsh vcpupin csr1000v 2 4

  virsh vcpupin csr1000v 3 5

  The host core number can be any number from 0 to 7. For more information, see the KVM documentation.

Note When you configure CPU pinning, consider the CPU topology of the host server. If you are using a CSR 1000v instance with multiple cores, do not configure CPU pinning across multiple sockets.

BIOS Settings

Optimize the performance of the KVM configuraiton by applying the recommended BIOS settings as mentioned in the following table:

Configuration	Recommended Setting
Intel Hyper-Threading Technology	Disabled
Number of Enable Cores	ALL
Execute Disable	Enabled
Intel VT	Enabled
Intel VT-D	Enabled
Intel VT-D coherency support	Enabled
Intel VT-D ATS support	Enabled
CPU Performance	High throughput
Hardware Perfetcher	Disabled
Adjacent Cache Line Prefetcher	Disabled
DCU Streamer Prefetch	Disable
Power Technology	Custom
Enhanced Intel Speedstep Technology	Disabled
Intel Turbo Boost Technology	Enabled
Processor Power State C6	Disabled
Processor Power State C1 Enhanced	Disabled
Frequency Poor Override	Enabled
P-State Coordination	HW_ALL
Energy Performance	Performance

For information about Red Hat Enterprise Linux requirements, see Bootstrap Properties, on page 123 and the subsequent sections.

Host OS Settings

In the host side, Cisco recommends that you use hugepages and enable emulator pinning. The following are some of the recommended settings in the host side:

- Enable IOMMU=pt
- Enable intel_iommu=on

- Enable hugepages
- Use SR-IOV if your system supports it for higher networking performance. Please check SR-IOV limitations your system might have.

In addition to enabling hugepages and emulator pinning, the following settings are also recommended: nmi_watchdog=0 elevator=cfq transparent_hugepage=never

Note If you use Virtio VHOST USER with VPP or OVS-DPDK, you can increase the buffer size to 1024 (rx_queue_size='1024') provided the version of your QEMU supports it.

IO Settings

You can use SR-IOV for better performance. However, note that this might bring in some limitations such as number of virtual functions (VF), OpenStack limitations for SR-IOV like QoS support, live migration and security group support.

If you use a modern vSwitch like fd.io VPP or OVS-DPDK, reserve at least 2 cores for the VPP worker threads or the OVS-DPDK PMD threads.

Configure the following parameters to run the VPP through command line:

- -cpu host: This parameter causes the VM to inherit the host OS flags. You require libvirt 0.9.11 or greater for this to be included in the xml configuration.
- -m 8192: You require 8GB RAM for optimal zero packet drop rates.
- rombar=0: To disable PXE boot delays, set rombar=0 to the end of each device option list or add "<rom bar=off />" to the device xml configuration.

Sample XMLs for KVM Performance Improvement

Sample XML for numa tuning

```
<numatune>
  <memory mode='strict' nodeset='0'/>
</numatune>
```

Sample XML for vCPU and emulator pinning

```
<cputune>
   <vcpupin vcpu='0' cpuset='3'/>
   <emulatorpin cpuset='3'/>
 </cputune>
```

Sample XML for hugepages

```
<currentMemory unit='KiB'>4194304</currentMemory>
 <memoryBacking>
   <hugepages>
     <page size='1048576' unit='KiB' nodeset='0'/>
```

```
    </hugepages>
    <nosharepages/>
  </memoryBacking>
```

Sample XML for virtio instead of IDE

```
<devices>
    <emulator>/usr/libexec/qemu-kvm</emulator>
    <disk type='file' device='disk'>
      <driver name='qemu' type='qcow2'/>
      <source file='/var/lib/libvirt/images/rhel7.0.qcow2'/>
      <backingStore/>
      <target dev='vda' bus='virtio'/>
      <boot order='1'/>
      <address type='pci' domain='0x0000' bus='0x00' slot='0x05' function='0x0'/>
    </disk>
```

Sample XML for VNC graphics

```
<graphics type='vnc' port='5900' autoport='yes' listen='127.0.0.1' keymap='en-us'>
      <listen type='address' address='127.0.0.1'/>
    </graphics>
```

XML for disabling memballon

```
<memballon model='none'>
```

Cloning the VM

In a KVM environment, cloning the Cisco CSR1000v virtual machine using the **virt-manager** virtual machine manager creates a Cisco CSR1000v virtual machine that may be un-bootable.

The issue is caused by an increase in the size of the cloned image size created by **virt-manager**, compared with the original Cisco CSR1000v VM image. The extra bytes (in the KB range) cause the boot failure.

Workarounds

There are three workarounds:

- Use the **virt-clone** command to clone the Cisco CSR 1000v VM image.
- For a cloned Cisco CSR 1000v VM image created by **virt-manager**, during the bootup, select the GOLDEN image to boot instead of packages.conf.
- In the “Create a new virtual machine” window, deselect “Allocate entire disk now” before the new Cisco CSR1000v VM is created. This ensures that the cloned Cisco CSR1000v VM image is able to boot up. However, this workaround does not support nested cloning. Use this method only on the first cloned Cisco CSR1000v VM image.

Configure the halt_poll_ns Parameter

On newer kernel releases (3.10.0-375.el7 and later), a KVM module parameter - halt_poll_ns has been introduced. You can use this parameter to alter the behaviour of how idle KVM guest virtual CPUs (vcpus) are handled.

When a virtual CPU in a KVM guest has no threads to run, the QEMU traditionally halts the idle CPU. This setting specifies a period of 400 nanoseconds by default, where a virtual CPU waits and polls before entering a CPU Idle state.

When new work arrives during the polling period before the vcpu is halted, the vcpu is immediately ready to execute the work. If the vcpu has been idle when new work arrives, the vcpu must be brought out of the idle state before the new work can be started. The time taken from idle to running state induces additional latency which negatively impacts latency sensitive workloads.

With the default kernel parameters, the guest Cloud Services Router (CSR1000v) CPU consumes 100% of the host CPU.

You can configure halt_poll_ns in two ways:

- Large halt_poll_ns: In this case, more CPU is spent busy-spinning for events that wake the virtual CPU, and less acpi deep sleeps occur. This means more power is consumed. However, there are less wakeups from deep states states, which depending on the state that's configured, can cause issues like cache misses etc.
- Small halt_poll_ns: In this case, less CPU time is spent busy-spinning for events that wake the CPU, more acpi deep sleeps occur. Here, less power consumed, but more wakeups from deep sleep states are required. More wakeups can cause large amounts of deep sleep instances, which depending on the configuration, can cause large amounts of cache misses and long wakeup time.

Configuring the halt_poll_ns parameter

You can configure the halt_poll_ns parameter in the following ways:

1. At run time, run the following: `echo 0 > /sys/module/kvm/parameters/halt_poll_ns`.
2. When you load the module, perform the following configuration:

```
# rmmod kvm_intel
# rmmod kvm
# modprobe kvm halt_poll_ns=0
# mpdprobe kvm_intel
```

3. When you boot the device, add `kvm.halt_poll_ns=<specify value>` in the parameters section of grub2.

Installation Requirements for KVM—Cisco IOS XE 3.x

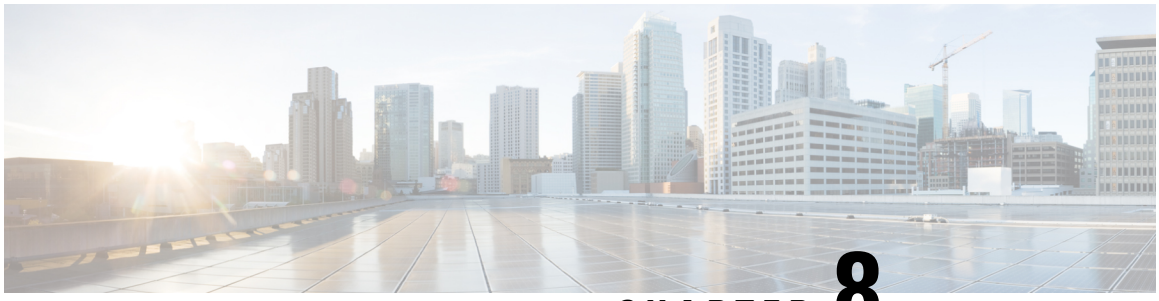
This section contains information about VMware requirements for older Cisco IOS XE releases (before IOS XE Denali 16.3.1).

The following table lists the installation requirements for KVM environments. For installation procedures, see Creating the Cisco CSR 1000v VM Using the Self-installing .Run Package, on page 113 and subsequent sections.

Table 28: Installation Requirements for KVM Environments (Cisco IOS XE versions 3.x)

KVM Requirements	Cisco IOS XE Release 3.10S	Cisco IOS XE Release 3.11S	Cisco IOS XE Release 3.12S and 3.13S	Cisco IOS XE Release 3.14S	Cisco IOS XE Release 3.15S, 3.16S, 3.17S
KVM versions supported	• Linux KVM based on Red Hat Enterprise Linux 6.3 (Requires Kernel version 2.6.32 and QEMU 0.12.1.2.) • Red Hat Enterprise Virtualization 3.1	• Linux KVM based on Red Hat Enterprise Linux 6.3 (Requires Kernel version 2.6.32 and QEMU 0.12.1.2.) • Red Hat Enterprise Virtualization 3.1 • Ubuntu 12.04.03 LTS Server 64 Bits(Requires QEMU-x86_64 version 1.0 (qemu-kvm-1.0))	• Linux KVM based on Red Hat Enterprise Linux 6.3 (Requires Kernel version 2.6.32 and QEMU 0.12.1.2.) • Ubuntu 12.04.04 LTS Server 64 Bits 2 (Requires QEMU-x86_64 version 1.0 (qemu-kvm-1.0))	• Linux KVM based on Red Hat Enterprise Linux 6.3 (Requires Kernel version 2.6.32 and QEMU 0.12.1.2.) • Ubuntu 14.04 LTS Server 64 Bits 2 (Requires QEMU-x86_64 version 1.0 (qemu-kvm-1.0))	• Linux KVM based on Red Hat Enterprise Linux 6.6 (Requires Kernel version 2.6.32-504 and QEMU 0.12.1.2.) (RHEL 7.1 (Requires Kernel version 3.10.0 and QEMU 1.5.3) supported in 3.17S) • Ubuntu 14.04 LTS Server 64 Bits 2 (Requires QEMU-x86_64 version 1.0 (qemu-kvm-1.0))
Supported vCPU configurations(The required vCPU configuration depends on the throughput license and technology package installed. See the data sheet for your release for more information.)	4 vCPUs: requires 4 GB RAM minimum allocation	• 1 vCPU: requires minimum 2.5 GB RAM allocation • 2 vCPUs: requires minimum 2.5 GB RAM allocation • 4 vCPUs: requires minimum 4 GB RAM allocation	• 1 vCPU: requires minimum 2.5 GB RAM allocation • 2 vCPUs: requires minimum 2.5 GB RAM allocation • 4 vCPUs: requires minimum 4 GB RAM allocation	• 1 vCPU: requires minimum 4 GB RAM allocation • 2 vCPUs: requires minimum 4 GB RAM allocation • 4 vCPUs: requires minimum 4 GB RAM allocation	• 1 vCPU: requires minimum 4 GB RAM allocation • 2 vCPUs: requires minimum 4 GB RAM allocation • 4 vCPUs: requires minimum 4 GB RAM allocation
Virtual CPU cores required	1	1	1	1	1

KVM Requirements	Cisco IOS XE Release 3.10S	Cisco IOS XE Release 3.11S	Cisco IOS XE Release 3.12S and 3.13S	Cisco IOS XE Release 3.14S	Cisco IOS XE Release 3.15S, 3.16S, 3.17S
Virtual hard disk size	8 GB minimum	8 GB minimum	8 GB minimum (Applies only to creating the VM using the .iso file. If using the .qcow2 file to install in an OpenStack environment, the hard disk size must be set to 0)	8 GB minimum (Applies only to creating the VM using the .iso file. If using the .qcow2 file to install in an OpenStack environment, the hard disk size must be set to 0)	8 GB minimum (Applies only to creating the VM using the .iso file. If using the .qcow2 file to install in an OpenStack environment, the hard disk size must be set to 0)
Supported vNICs	Virtio	Virtio	Virtio	Virtio	Virtio
Maximum number of vNICs supported per VM instance	26	26	26	26	26
Virtual CD/DVD drive installed (applicable only when installing using an .iso file)	Required	Required	Required	Required	Required



CHAPTER 8

Installing the Cisco CSR 1000v in Microsoft Hyper-V Environments

Microsoft Hyper-V Support Information

(Cisco IOS XE Release 3.12S or later and Cisco IOS XE Release Denali 16.3.1 or later)—the Cisco CSR 1000v supports installation on Microsoft Hyper-V using Windows Server 2012 R2.

Note (Cisco IOS XE Denali 16.2)—Installing Cisco CSR 1000v in a Microsoft Hyper-V environment is not supported.

The Cisco CSR 1000v installation on Microsoft Hyper-V requires the manual creation of a VM and installation using the .iso file. Deploying the OVA template into a Microsoft Hyper-V environment is not supported.

The following Microsoft Hyper-V features are supported:

- Live Migration
- Snapshot
- Move
- Export
- Hyper-V Replica

For more information about Microsoft Hyper-V, see the Microsoft Windows Server 2012 R2 documentation.

Microsoft Hyper-V Limitations

This section describes the limitations when specifying VLANs on a VM interface, using the Hyper-V Manager.

(Cisco IOS XE Denali 16.3.1 or later) You can only add one VLAN for a VM interface using the Virtual Switch Manager of Hyper-V Manager.

(Cisco IOS XE 3.17 or earlier) The Cisco CSR 1000v vNIC/interface numbering may change after the Cisco CSR 1000v (running on Hyper-V) is replicated or migrated to another server. To prevent the interface numbering from changing, as a workaround, execute the `clear platform software vnic-if nvtable` command before the failover/restart occurs.

(Cisco IOS XE 3.17 or earlier) You can only add one VLAN for a VM interface using the Virtual Switch Manager of Hyper-V Manager. However, using Cisco IOS XE 3.17 or earlier, you also have the option of using a powershell CLI command **Set-VMNetworkAdapterVlan** to specify multiple VLANs. See the following example:

Set-VMNetworkAdapterVlan -VMName dr-vm-6-1 **-Trunk -AllowedVlanIdList** 1-300 **-NativeVlanId** 0

Note The **Set-VMNetworkAdapterVlan** command must be re-entered every time that the Cisco CSR 1000v is reloaded. We recommend that limit the number of VLANs to 300 or below—using the **AllowedVlanIdList** parameter.

For more information on the **Set-VMNetworkAdapterVlan** powershell command, see https://technet.microsoft.com/itpro/powershell/windows/hyper-v/set-vmnetworkadaptervlan.

See also Configure virtual local area networks for Hyper-V.

Installation Requirements for Microsoft Hyper-V

The Microsoft Hyper-V requirements for Cisco CSR 1000V using Cisco IOS XE 16.12 through 17.3 releases are as follows:

- The following Microsoft Hyper-V versions are supported:
  - Windows Server 2016 is recommended - tested and meets the performance benchmarks.
- vCPUs. The following vCPU configurations are supported:
  - 1 vCPU: requires minimum 4 GB RAM allocation
  - 2 vCPUs: requires minimum 4 GB RAM allocation
  - 4 vCPUs: requires minimum 4 GB RAM allocation
- Virtual CPU cores—1 vCPU is required
- Virtual hard disk size—8 GB minimum
- Supported vNICs—NetVSC (pmap)

- Maximum number of vNICs supported per VM instance—8
- Virtual CD/DVD drive installed—required

Note The Microsoft Hyper-V requirements for older versions of Cisco IOS XE (before IOS XE Denali 16.3) are shown in Installation Requirements for Microsoft Hyper-V—Cisco IOS XE 3.x, on page 142.

Manually Creating the Cisco CSR 1000v VM using the .iso File (Microsoft Hyper-V)

Prerequisites

Prerequisites for Manually Creating the CSR 1000v VM using the .iso File

While the following procedure provides a general guideline for how to manually create the VM for the Cisco CSR 1000v, the exact steps that you need to perform may vary depending on the characteristics of your Microsoft Hyper-V environment and setup. For more information, see Microsoft Windows Server 2012 R2 documentation.

Note The Cisco CSR 1000v does not support deploying the OVA file in Microsoft Hyper-V environments.

Before installing the Cisco CSR 1000v on a Microsoft Hyper-V VM, the following must be installed on the host:

- Hyper-V Manager
- Failover Cluster Manager
- Virtual Switch

Although not required, it is recommended that you create the Virtual Switch prior to creating the VM for the Cisco CSR 1000v.

Configuring the Server Manager Settings

The following steps are performed on Server Manager on the host before creating the Cisco CSR 1000v VM.

SUMMARY STEPS

1. On the Server Manager, select **Dashboard** to configure the local server.
2. Select **Manager** from the top right, and then select **Add Roles and Features** from the drop-down menu.
3. Click **Next**.
4. Select **Server Roles**. In the Roles list, select the following options by clicking on the checkbox:
5. Select Features. In the Features list, select the following option by clicking on the checkbox:

6. Click **Next**.

DETAILED STEPS

Procedure

Step 1 On the Server Manager, select **Dashboard** to configure the local server.

Step 2 Select **Manager** from the top right, and then select **Add Roles and Features** from the drop-down menu.

The **Add Roles and Features Wizard opens.**

Step 3 Click **Next**.

Step 4 Select **Server Roles**. In the Roles list, select the following options by clicking on the checkbox:

- **File and Storage Services**
- **Hyper-V**

Step 5 Select Features. In the Features list, select the following option by clicking on the checkbox:

- Failover Clustering

Failover clustering is required. It is not automatically installed, so you must make sure this option is checked. This feature requires that Failover Cluster Manager is installed.

Step 6 Click **Next**.

Creating the VM

To create the VM, perform the following steps:

SUMMARY STEPS

1. In Hyper-V Manager, click on the host.
2. Select **New > Virtual Machine**.
3. Click **Specify Name and Location.**
4. On the **Assign Memory** screen, enter the **Startup Memory** value.
5. On the **Configure Networking** screen, select a network connection to the virtual switch that was previously created.
6. On the **Connect Virtual Hard Disk Screen**, select the following option:
7. Review the VM settings, and if correct, click **Finish**.

DETAILED STEPS

Procedure

Step 1 In Hyper-V Manager, click on the host.

Step 2 Select **New > Virtual Machine**.

Step 3 Click **Specify Name and Location.**

- Enter the name of the VM.
- (Optional) Click the checkbox to store the VM in a different location.

Click **Next**.

Step 4 On the **Assign Memory** screen, enter the **Startup Memory** value.

The Cisco CSR 1000v requires 4096 MB for the startup memory.

Click **Next**.

Step 5 On the **Configure Networking** screen, select a network connection to the virtual switch that was previously created.

The network adapter selected in this step will become the first interface for the Cisco CSR 1000v once the VM is launched and the router boots. The other vNICs for the VM are created in the next procedure.

Note
Changing the MAC address of the first interface and rebooting a licensed Cisco CSR 1000v will de-activate the license.

Click **Next**.

Step 6 On the **Connect Virtual Hard Disk Screen**, select the following option:

- Attach a virtual hard disk later.

Note
The New Virtual Machine Wizard only supports creating a virtual hard disk using the .vhdx format. The Cisco CSR 1000v requires that the hard disk uses the .vhd format. You will create the virtual hard disk after the VM has been created.

Click **Next**. The **Summary** screen displays.

Step 7 Review the VM settings, and if correct, click **Finish**.

The new VM is created.

Configuring the VM Settings

To configure the VM settings before launching the VM, perform the following steps:

SUMMARY STEPS

1. In Hyper-V Manager, select the host, and then right-click on the VM that was created in the previous steps.

2. Select **Settings**.
3. Specify the number of virtual processors, also known as virtual CPU's (vCPU's) for the VM.
4. Under IDE Controller 0, select the Hard Drive.
5. Under IDE Controller1, select the **DVD Drive**.
6. Select **Network Adapte**r to verify that the network connection to the virtual switch is configured.
7. Select **Com 1** to configure the serial port.
8. Select **Hardware** > **Add Hardware** to add the network interfaces (vNICs) to the VM.
9. Click **BIOS** to verify the boot sequence for the VM.

DETAILED STEPS

Procedure

Step 1 In Hyper-V Manager, select the host, and then right-click on the VM that was created in the previous steps.

Step 2 Select **Settings**.

Step 3 Specify the number of virtual processors, also known as virtual CPU's (vCPU's) for the VM.

See table "Installation Requirements for Microsoft Hyper-V" below, for the supported configurations.

Step 4 Under IDE Controller 0, select the Hard Drive.

Click the **Virtual Hard Disk** checkbox and click **New** to create a new virtual hard disk.

The New Virtual Hard Disk Wizard opens. Click **Next**.

a) On the Choose Disk Format screen, click the VHD checkbox to create the virtual hard disk using the .vhd format. Click **Next**.

Note
The Cisco CSR 1000v does not support the VHDX format.

b) On the **Choose Disk Type** screen, click on the **Fixed Size** option. Click **Next**.

The Cisco CSR 1000v does not support the other disk type options.

c) Specify the Name and Location for the virtual hard disk. Click **Next**.
d) On the **Configure Disk** screen, click the option to create a new blank virtual hard disk. For the size, specify 8 GB.
e) Click **Next** to view the Summary of the virtual hard disk settings.
f) Click **Finish** to create the new virtual hard disk.

When the new hard disk has been created, continue configuring the VM settings with the next step.

Step 5 Under IDE Controller1, select the **DVD Drive**.

The DVD Drive screen displays.

For the **Media** setting, click the **Image File** checkbox, and browse to the Cisco CSR 1000v .iso file that you downloaded from Cisco.com.

Click **OK**.

Step 6 Select **Network Adapte**r to verify that the network connection to the virtual switch is configured.

Step 7 Select **Com 1** to configure the serial port.

This port provides access to the Cisco CSR 1000v console.

Note
Telnet access to the Cisco CSR 1000v console is not supported for Microsoft Hyper-V. You must use a Putty session to access the console.

Step 8 Select **Hardware** > **Add Hardware** to add the network interfaces (vNICs) to the VM.

a) Select **Network Adapter** and click **Add**.

Microsoft Hyper-V adds the network adapter and highlights that hardware with the status Virtual Switch "Not Connected".

b) Select a virtual switch on the drop-down menu to place the network adapter onto it.

Repeat these steps for each vNIC added. The Cisco CSR 1000v supports only the HV NETVSC vNIC type. The maximum number of vNICs supported is 8.

Note
The hot-add of vNICs is not supported with Microsoft Hyper-V, so the network interfaces need to be added before launching the VM.

After the Cisco CSR 1000v boots, you can verify the vNICs and how they are mapped to the interfaces using the **show platform software vnic-if interface-mapping** command. See Mapping Cisco CSR 1000v Network Interfaces to VM Network Interfaces, on page 251.

Step 9 Click **BIOS** to verify the boot sequence for the VM.

The VM should be set to boot from the CD.

Launching the VM to Boot the Cisco CSR 1000v

To launch the VM, perform the following steps:

SUMMARY STEPS

1. Select the virtual switch.
2. Select the VM and click **Start**.

DETAILED STEPS

Procedure

Step 1 Select the virtual switch.

Step 2 Select the VM and click **Start**.

The Hyper-V Manager connects to the VM, and starts the launch process. Once the VM is launched, the Cisco CSR 1000v starts the boot process. For more information on the booting process, see Installing the Cisco CSR 1000v in Microsoft Hyper-V Environments, on page 135.

Installation Requirements for Microsoft Hyper-V—Cisco IOS XE 3.x

This section contains information about Microsoft Hyper-V requirements for older Cisco IOS XE releases (the releases supported by Cisco CSR 1000v before IOS XE Denali 16.3.1)

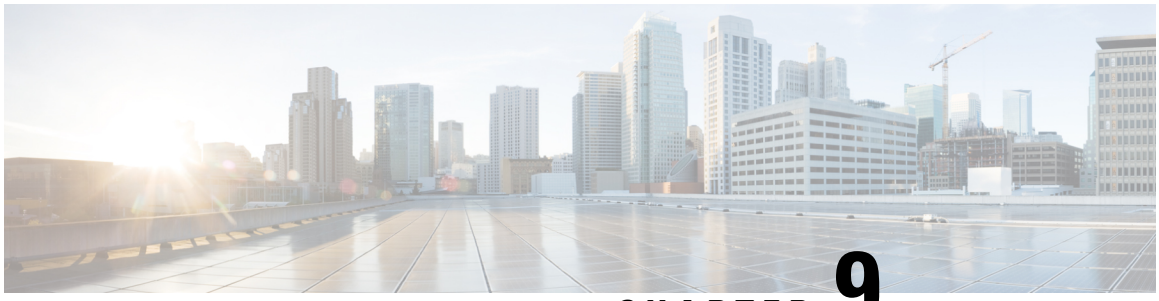
The table below lists the installation requirements for Microsoft HyperV.

Table 29: Installation Requirements for Microsoft Hyper-V (Cisco IOS XE versions 3.x)

Microsoft Hyper-V Requirements	Cisco IOS XE 3.12S and 3.13S	Cisco IOS XE 3.14S, 3.15S, 3.16S, 3.17
Microsoft Hyper-V version supported	Windows Server 2012 R2	Windows Server 2012 R2
Supported vCPU configurations[10]	• 1 vCPU: requires minimum 2.5 GB RAM allocation • 2 vCPUs: requires minimum 2.5 GB RAM allocation • 4 vCPUs: requires minimum 4 GB RAM allocation	• 1 vCPU: requires minimum 4 GB RAM allocation • 2 vCPUs: requires minimum 4 GB RAM allocation • 4 vCPUs: requires minimum 4 GB RAM allocation
Virtual CPU cores required	1	1
Virtual hard disk size[11]	8 GB	8 GB
Supported vNICs	HV driver	HV driver
Maximum number of vNICs supported per VM instance	8	8
Virtual CD/DVD drive Installed	Required	Required

[10] The required vCPU configuration depends on the throughput license and technology package installed. See the data sheet for your release for more information.

[11] The VHD format is supported only. The VHDX format is not supported.



CHAPTER 9

Booting the Cisco CSR 1000v and Accessing the Console

Booting the Cisco CSR 1000v as the VM

The Cisco CSR 1000v boots when the VM is powered on. Depending on your configuration, you can monitor the installation process on the virtual VGA console or the console on the virtual serial port.

Note If you want to access and configure the Cisco CSR 1000v from the serial port on the hypervisor instead of the virtual VGA console, you should provision the VM to use this setting before powering on the VM and booting the router. For more information, see theIntroduction to Accessing the Cisco CSR 1000v through the Virtual Serial Port, on page 146

SUMMARY STEPS

1. Power-up the VM. Within 5 seconds of powering on the VM, choose a console described from one of the following three steps (2, 3,or 4) to select a console to view the router bootup and to access the Cisco CSR 1000v CLI.
2. (Optional) Select **Auto Console**: (Cisco IOS XE 3.13S and later, and IOS XE Denali 16.3.1 and later)
3. (Optional) Select **Virtual Console**
4. (Optional)Select **Serial Console**
5. Telnet to the VM using one of the following two commands: **telnet://***host-ipaddress***:***portnumber* or, from a UNIX xTerm terminal: **telnet** *host-ipaddress portnumber* The following example shows the Cisco CSR 1000v initial boot output on the VM.
6. After booting, the system presents a screen showing the main software image and the Golden Image, with an instruction that the highlighted entry is booted automatically in three seconds. Do not select the option for the Golden Image and allow the main software image to boot.

DETAILED STEPS

Procedure

Step 1 Power-up the VM. Within 5 seconds of powering on the VM, choose a console described from one of the following three steps (2, 3,or 4) to select a console to view the router bootup and to access the Cisco CSR 1000v CLI.

Step 2 (Optional) Select **Auto Console**: (Cisco IOS XE 3.13S and later, and IOS XE Denali 16.3.1 and later)

Choose this option to use automatic console detection. When two virtual serial ports are detected, the IOS XE CLI will be available on the first virtual serial port and the IOS XE diagnostic CLI will be available on the second virtual serial port. If two virtual serial ports are not detected, the IOS XE CLI will be available on the virtual VGA console. This is the default setting and the Cisco CSR 1000v will boot using the automatic console detection if another option is not selected within the 5 second timeframe.

Note (for **VMware ESXi**): If you are installing on VMware ESXi without a virtual serial port concentrator (vSPC), this option may not be able to properly detect virtual serial ports when there is an active connection to the virtual serial ports. If you are not using a vSPC and wish to use virtual serial ports, choose the Serial Console option.

Note (for **Microsoft Hyper-V**): If you are installing on Microsoft Hyper-V, this option may be unable to properly detect virtual serial ports when there is an active connection to the virtual serial ports. If you wish to use virtual serial ports, you should choose the Serial Console option.

(Optional)**Automatic selection of virtual serial** portsFor this option, the virtual serial ports must already be present on the VM. The virtual serial port must already be present on the VM for this option to work.

If you are installing on VMware ESXi, see Creating Serial Console Access in VMware ESXi, on page 146.

If you are installing in KVM environments, see Creating the Serial Console Access in KVM, on page 148.

If you are installing in Microsoft Hyper-V environments, see Creating the Serial Console Access in Microsoft Hyper-V, on page 149.

The Cisco CSR 1000v starts the boot process.

Step 3 (Optional) Select **Virtual Console**

Choose this option to use the virtual VGA console. If you choose to use the virtual console, the rest of the steps in this procedure do not apply. On Cisco IOS XE 3.12S and earlier, this is the default setting and the Cisco CSR 1000v boots using the Virtual Console if another option is not selected within the 5 second timeframe.

The Cisco CSR 1000v starts the boot process.

Step 4 (Optional)Select **Serial Console**

Choose this option to use the virtual serial port console on the VM (not supported on Citrix XenServer VMs).

The virtual serial port must already be present on the VM for this option to work.

If you are installing on VMware ESXi, see Creating Serial Console Access in VMware ESXi, on page 146.

If you are installing in KVM environments, see Creating the Serial Console Access in KVM, on page 148.

If you are installing in Microsoft Hyper-V environments, see Creating the Serial Console Access in Microsoft Hyper-V, on page 149.

Note

The option to select the console port during the boot process is available only the first time the Cisco CSR 1000v boots. To change the console port access after the Cisco CSR 1000v has first booted, see Changing the Console Port Access After Installation, on page 150 the “Changing the Console Port Access After Installation” section on page 8-7.

The Cisco CSR 1000v starts the boot process.

Step 5 Telnet to the VM using one of the following two commands: **telnet://***host-ipaddress***:***portnumber* or, from a UNIX xTerm terminal: **telnet** *host-ipaddress portnumber* The following example shows the Cisco CSR 1000v initial boot output on the VM.

Example:

```
%IOSXEBOOT-4-BOOT_SRC: (rp/0): CD-ROM Boot%IOSXEBOOT-4-BOOT_CDROM: (rp/0):
Installing GRUB%IOSXEBOOT-4-BOOT_CDROM: (rp/0): Copying super packagecsr1000v-universalk9
2011-10-20_13.09.SSA.bin%IOSXEBOOT-4-BOOT_CDROM: (rp/0):
Creating /boot/grub/menu.lst%IOSXEBOOT-4-BOOT_CDROM: (rp/0):
CD-ROM Installation finished%IOSXEBOOT-4-BOOT_CDROM: (rp/0): Ejecting CD-ROM tray
```

The system first calculates the SHA-1, which may take a few minutes. Once the SHA-1 is calculated, the kernel is brought up. Once the initial installation process is complete, the .iso package file is removed from the virtual CD-ROM, and the VM is rebooted. This enables the Cisco CSR 1000v to boot normally off the virtual Hard Drive.

Note
The system reboots during first-time installation only.

The time required for the Cisco CSR 1000v to boot may vary depending on the release and the hypervisor used.

Step 6 After booting, the system presents a screen showing the main software image and the Golden Image, with an instruction that the highlighted entry is booted automatically in three seconds. Do not select the option for the Golden Image and allow the main software image to boot.

Note
The Cisco CSR 1000v does not include a ROMMON image that is included in many Cisco hardware-based routers. During installation, a “backup” copy of the installed version is stored in a backup partition. This copy can be selected to boot from in case you upgraded your boot image, deleted the original boot image, or somehow corrupted your disk. Booting from the backup copy is equivalent to booting a different image from ROMMON.For more information on changing the configuration register settings to access GRUB mode, see Accessing and Using GRUB Mode, on page 263.

You can now enter the router configuration environment by entering the standard commands **enable** and then **configure terminal. The following should be noted for the initial installation:**

When the Cisco CSR 1000v is booted for the first time, the mode the router boots in depends on the release version.

For Cisco IOS XE 3.13S and later, and IOS XE Denali 16.3.1 and later, the Cisco CSR 1000v boots with the AX package set of features and throughput is limited to 100 Kbps. For Cisco IOS XE 3.12S and earlier, the Cisco CSR 1000v boots in a limited mode that provides limited feature support and throughput is limited to 2.5 Mbps.

You must install the software license or enable an evaluation license to obtain the supported throughput and features. Depending on the release version, you must enable the boot level or change the maximum throughput level, and reboot the Cisco CSR 1000v. For more information, see Installing Cisco CSR 1000v Licenses, on page 161.

For Cisco IOS XE 3.13S and later, and IOS XE Denali 16.3.1 and later, the installed license technology package must match the package level configured with the **license boot level** command. If the license package does not match the configured setting, throughput is limited to 100 Kbps.

(VMware ESXi only) If you manually created the VM using the .iso file, then you need to configure the basic router properties. You can use either the Cisco IOS XE CLI commands or you can manually configure the properties in the vSphere GUI. For more information, see Editing the Basic Properties of Cisco CSR 1000v using vSphere, on page 85.

Accessing the Cisco CSR 1000v Console

Accessing the Cisco CSR 1000v Through the Virtual VGA Console

When installing the Cisco CSR 1000v software image, the setting to use is as follows:

- (Cisco IOS XE 3.12S and earlier, Cisco_IOS XE 3.17S and later, and IOS XE Denali 16.3.1 and later) Virtual VGA console
- (Cisco IOS XE 3.12S through Cisco IOS XE 3.16S) Automatic console detection

No other configuration changes are required to access the Cisco CSR 1000v CLI through the virtual VGA console if:

- You do not change the console setting during the bootup process

and

- (If using automatic console detection) You do not add two virtual serial ports to the VM configuration

Accessing the Cisco CSR 1000v Through the Virtual Serial Port

Introduction to Accessing the Cisco CSR 1000v through the Virtual Serial Port

By default, the Cisco CSR 1000v is accessed using the virtual VGA console. If using automatic console detection and two virtual serial ports are detected, the Cisco CSR 1000v CLI will be available on the first virtual serial port.

You can also configure the VM to use the Serial Console, which always attempts to use the first virtual serial port for the Cisco CSR 1000v CLI. See the following sections to configure the virtual serial port on your hypervisor.

Note The Citrix XenServer does not support access through a serial console.

Creating Serial Console Access in VMware ESXi

Perform the following steps using VMware VSphere. For more information, refer to the VMware VSphere documentation.

SUMMARY STEPS

1. Power-down the VM.

2. Select the VM and configure the virtual serial port settings.
3. Select **Select Network Backing**.
4. Power on the VM.
5. When the VM is powered on, access the virtual serial port console.
6. Configure the security settings for the virtual serial port.

DETAILED STEPS

Procedure

Step 1 Power-down the VM.

Step 2 Select the VM and configure the virtual serial port settings.

a) Choose Edit Settings > Add.

b) Choose Device Type > Serial port.

Click **Next**.

c) Choose **Select Port Type**.

Select **Connect via Network** and click **Next**.

Step 3 Select **Select Network Backing**.

Select the **Server (VM listens for connection)** option.

Enter the **Port URI** using the following syntax:

telnet://:*portnumber*

where *portnumber* is the port number for the virtual serial port.

Under I/O mode, select the option **Yield CPU on poll** and click **Next**.

Step 4 Power on the VM.

Step 5 When the VM is powered on, access the virtual serial port console.

Step 6 Configure the security settings for the virtual serial port.

a) Select the ESXi host for the virtual serial port.

b) Click the **Configuration** tab and click **`Security Profile`**.

c) In the Firewall section, click **Properties**, and then select the **VM serial port connected over Network** value.

You can now access the Cisco IOS XE console using the Telnet port URI. When you configure the virtual serial port, the CSR 1000v is no longer accessible from the VM's virtual console. See Opening a Telnet Session to the Cisco CSR 1000v Console on the Virtual Serial Port, on page 149.

Note
To use these settings, either the **Auto Console** option or the **Serial Console** option in the GRUB menu must have been selected during the Cisco CSR 1000v bootup. If you have already installed the Cisco CSR 1000v software using the virtual VGA console, you must configure either the Cisco IOS XE **platform console auto command or t**he Cisco IOS XE **platform console serial command** and reload the VM for the console access through the virtual serial port to work. See Changing the Console Port Access After Installation, on page 150.

Creating the Serial Console Access in KVM

Perform the following steps using the KVM console on your server. For more information, refer to the KVM documentation.

SUMMARY STEPS

1. Power off the VM.
2. Click on the default **Serial 1** device (if it exists) and then click **Remove**. This removes the default pty-based virtual serial port which would otherwise count as the first virtual serial port.
3. Click **Add Hardware**.
4. Select **Serial** to add a serial device.
5. Under Character Device, choose the **TCP Net Console (tcp)** device type from the drop-down menu.
6. Under Device Parameters, choose the mode from the drop-down menu.
7. Under Host, enter 0.0.0.0. The server will accept a telnet connection on any interface.
8. Choose the port from the drop-down menu.
9. Choose the **Use Telnet** option.
10. Click **Finish**.

DETAILED STEPS

Procedure

Step 1 Power off the VM.

Step 2 Click on the default **Serial 1** device (if it exists) and then click **Remove**. This removes the default pty-based virtual serial port which would otherwise count as the first virtual serial port.

Step 3 Click **Add Hardware**.

Step 4 Select **Serial** to add a serial device.

Step 5 Under Character Device, choose the **TCP Net Console (tcp)** device type from the drop-down menu.

Step 6 Under Device Parameters, choose the mode from the drop-down menu.

Step 7 Under Host, enter 0.0.0.0. The server will accept a telnet connection on any interface.

Step 8 Choose the port from the drop-down menu.

Step 9 Choose the **Use Telnet** option.

Step 10 Click **Finish**.

You can now access the Cisco IOS XE console using the Telnet port URI. See the Opening a Telnet Session to the Cisco CSR 1000v Console on the Virtual Serial Port, on page 149.

Note
To use these settings, either the **Auto Console** option or the **Serial Console** option in the GRUB menu must have been selected while the Cisco CSR 1000v booted. If you have already installed the Cisco CSR 1000v software using the virtual VGA console, you must configure either the Cisco IOS XE **platform console auto** command or **platform console serial command** and reload the VM in order for the console access through the virtual serial port to work. See the Changing the Console Port Access After Installation, on page 150.

Creating the Serial Console Access in Microsoft Hyper-V

The console port access for Microsoft Hyper-V is created when configuring the VM settings. For more information, see the "Configuring the VM Settings" section on page 7-4 .

Note Telnet access to the Cisco CSR 1000v console is not supported for Microsoft Hyper-V. You must use a Putty session to access the console.

Opening a Telnet Session to the Cisco CSR 1000v Console on the Virtual Serial Port

Perform the following steps using the Cisco IOS XE CLI commands:

SUMMARY STEPS

1. Telnet to the VM
2. At the Cisco CSR 1000v IOS XE password prompt, enter your login password. The following example shows entry of the password *mypass*:
3. From user EXEC mode, enter the **enable** command as shown in the following example:
4. At the password prompt, enter your system password. The following example shows entry of the password *enablepass*:
5. When the enable password is accepted, the privileged EXEC mode prompt appears:
6. You now have access to the CLI in privileged EXEC mode and you can enter the necessary commands to complete your desired tasks.
7. To exit the Telnet session, use the **exit** or **logout** command as shown in the following example:

DETAILED STEPS

Procedure

Step 1 Telnet to the VM

- Use the following command **telnet://***host-ipaddress***:***portnumber*
- Or, from a UNIX terminal use the command

  telnet *host-ipaddress portnumber*

Step 2 At the Cisco CSR 1000v IOS XE password prompt, enter your login password. The following example shows entry of the password *mypass*:

Example:

```
User Access Verification
Password: mypass
```

Note
If no password has been configured, press **Return**.

Step 3 From user EXEC mode, enter the **enable** command as shown in the following example:

Example:

```
Router> enable
```

Step 4 At the password prompt, enter your system password. The following example shows entry of the password *enablepass*:

Example:

```
Password: enablepass
```

Step 5 When the enable password is accepted, the privileged EXEC mode prompt appears:

Example:

```
Router#
```

Step 6 You now have access to the CLI in privileged EXEC mode and you can enter the necessary commands to complete your desired tasks.

Step 7 To exit the Telnet session, use the **exit** or **logout** command as shown in the following example:

Example:

```
Router# logout
```

Changing the Console Port Access After Installation

After the Cisco CSR 1000v has booted successfully, you can change the console port access to the router using Cisco IOS XE commands. After you change the console port access, you must reload or power-cycle the router.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. Do one of the following:
   - **platform console auto**
   - **platform console virtual**
   - **platform console serial**
4. **end**
5. **copy system:running-config nvram:startup-config**
6. **reload**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.

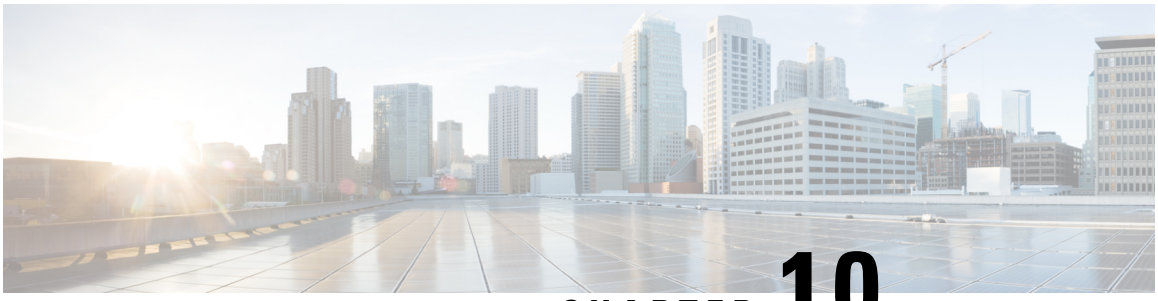
	Command or Action	Purpose
	Example: `Router> enable`	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.

Step 3	Do one of the following:
	• **platform console auto**
	• **platform console virtual**
	• **platform console serial**
	Example:
	`Router(config)# platform console auto`
	Example:
	`Router(config)# platform console virtual`
	Example:
	`Router(config)# platform console serial`
		Options for **platform console** *x*:
		• **auto**—Specifies that the Cisco CSR 1000v console is detected automatically. This is the default setting during the initial installation boot process (Cisco IOS XE 3.13S and later). For additional information, see Booting the Cisco CSR 1000v as the VM, on page 143.
		• **virtual**—Specifies that the Cisco CSR 1000v is accessed through the hypervisor virtual VGA console. This is the default setting during the initial installation boot process (on Cisco IOS XE 3.12S and earlier).
		• **serial**—Specifies that the Cisco CSR 1000v is accessed through the serial port on the VM.
		Note: Use this option only if your hypervisor supports serial port console access.

Step 4	**end** **Example:** `Router(config)# end`	Exits configuration mode.
Step 5	**copy system:running-config nvram:startup-config** **Example:** `Router# copy system:running-config nvram:startup-config`	Copies the running configuration to the NVRAM startup configuration.
Step 6	**reload** **Example:** `Router# reload`	Reloads the operating system.

License Installation

One of the first steps you need to perform after obtaining console access is to install the Cisco CSR 1000v software licenses. For more information, see Installing Cisco CSR 1000v Licenses, on page 161.



CHAPTER 10

Day 0 Configuration For CSR 1000v Release 17.2 and Later

Information About Day0 Configuration

The Cisco CSR 1000v instance requires manual configuration before the device is fully functional. To automate the configuration steps or to connect to on-premise sites, you can upload the CSR 1000v custom data or user data in all the supported public and private clouds.

By uploading the custom data for your cloud service provider or your private cloud, you can automate the day 0 and/or the bootstrap configuration. Upload or attach a bootstrap configuration file, (iosxe_config.txt file, ciscosdwan_cloud_init.cfg file or a ciscosdwan.cfg file) or provide the user data to automate these processes to bring up the device into a functional state with minimal to no touch.

Starting Cisco IOS XE Release 17.2, the Day0 configuration is changed. The same universalk9 image is available to deploy Cisco IOS XE (autonomous mode) and Cisco IOS XE SD-WAN (controller mode) features on Cisco IOS XE devices. After you deploy the CSR 1000v Release 17.2 image, the Day0 and/or the bootstrap configuration is used to determine if the router has to boot up in the Controller mode or the autonomous mode.

Autonomous and Controller Mode

You can access the Cisco IOS XE and the Cisco IOS XE SD-WAN functionalities by choosing either the autonomous mode or the Controller mode, respectively. The autonomous mode is the default mode for Cisco CSR 1000v and includes the Cisco IOS XE functionalities. To access the Cisco IOS XE SD-WAN functionalities, switch to the controller mode.

The following are the main differences between autonomous mode and controller modes:

Table 30:

Feature	Autonomous Mode	Controller Mode
Configuration method	• CLI • NETCONF	• YANG-based configuration • Cisco vManage • NETCONF

Feature	Autonomous Mode	Controller Mode
Onboarding modes	• Config-Wizard • WebUI • USB • Auto-install (Python script, TCL script) • ZTP (using DCHP option 150 and option 67)	Plug and Play USB
Interconnectivity	Network interface	VPN
Licensing	• Cisco Smart Licensing • PayG	Cisco High Performance Security (HSEC) software licensing. No device licensing.
Dual-IOSd redundancy model	Supported	Not supported
High availability	Supported	Not supported
Global configuration mode	**configure terminal** command	**configure transition** command

Note When you upgrade the CSR 1000v image to the IOS XE 17.2 release or later, if the system is unable to detect any of the following four parameters – OTP, UUID, VBOND, ORG, the device boots in the autonomous mode.

To switch between controller and autonomous modes, see the *Switching Between Autonomous and Controller Modes* section in this feature document.

If you are a user who wants to proceed with the autonomous mode configuration, continue reading this feature document. If you wish to deploy the CSR 1000v instance in the controller mode, see Install and Upgrade for Cisco IOS 17.2 and Later.

Prerequisites for Deploying the Unified Image

- Download the install the CSR 1000v 17.2 image. For more information, see CSR1000v Installation Overview.

- If you want to deploy the CSR 1000v instance in the controller mode, generate the bootstrap config file from vManage.

Restrictions for Deploying the Unified Image

- Starting from Cisco IOS XE Release 17.2, ucmk9 image is not published.
- If you use the PayG licensing model, you cannot perform a mode switch as controller mode does not support the PayG licensing model.
- Only the autonomous mode supports Dual-IOSd.
- Images without payload encryption and NO-LI images are not supported in the controller mode.
- After onboarding and determining the mode of operation, if you switch from the controller mode to the autonomous mode or the reverse, results in loss of configuration.
- When you switch from the autonomous mode to the controller mode or vice versa, your Smart Licensing registration does not work. You must reregister for your Smart Licenses to work.

How to Perform Day0 Configuration

Bootstrap Configuration Files

On a device that already runs a Cisco IOS XE non-SDWAN image, after you install the Cisco IOS XE

17.2 image, when you launch the CSR 1000v instance for the first time, in the absence of bootstrap configuration the instance always comes up in the autonomous mode. If you provide any user-data or custom-data or bootstrap configuration to the instance depending on the cloud environment, the data is used for the bootstrap configuration. To know more about the Day0 or bootstrap configuration for each service provider, see Day0 and Custom Data Configuration in this feature document.

On a new, out of box device, if you want to boot up the device in the autonomous mode, you need not provide the bootstrap configuration. In this scenario, by default, the instance always boots up in the autonomous mode. If you want to provide bootstrap related configurations, upload the iosxe_config.txt file or the ovf-env.xml file.

Note In the case of public clouds, the filename does not matter as the instance fetches the latest user data or the custom data from metadata. However, in the case of private clouds, if you upload the .iso file, the filename is important.

On a new, out of box device, if you want to boot up the device in the controller mode, make sure that all the four parameters (OTP, UUID, VBOND, ORG) is present in the `ciscosdwan.cfg/ciscosdwan_cloud_init.cfg` file for a fresh deployment on Cisco CSR1000v or Cisco ISRv devices. After the device boots up in the controller mode, the configuration present in the configuration file is applied.

Day 0 and Custom Data Configuration for the Cloud Service Providers

Based on the cloud in which you are deploying the CSR 1000v instance, see the following to perform the bootstrap and/or the day 0 configuration:

- Deploying the OVA to the VM
- Manually creating the Cisco CSR 1000v VM using the .iso file (Citrix XenServer)
- Creating a CSR 1000v VM using the self installing .run package
- Manually creating the VM using the .iso file (Microsoft Hyper-V)
- Booting the CSR 1000v Instance
- Deploying a CSR 1000v VM Using Custom Data
- Deploying a CSR 1000v VM on Microsoft Azure

Note For a CSR 1000v instance running on Cisco CSP-5000 hypervisor, when you enter the settings in the Day Zero Config screen, ensure that you maintain the format mentioned here:

- **Source File Name**: Enter the value for this field in the format: *day0_filename cisco_sdwan.cfg*.
- **Destination File Name**: Enter the value for this field in the format: *day0-dest-filename /openstack/content/cisco_sdwan.cfg*.

Verifying the Router Operation Mode and Day 0 Configuration

To verify whether you've deployed or upgraded to the IOS XE 17.2 release successfully, run the **show version** command. The **operating device-mode** parameter displays whether the CSR 1000v instance is running in the autonomous or the controller mode.

Sample configuration output for CSR1000v instance in autonomous mode

```
Device# show version | inc operating
Router operating mode: Autonomous
Device# show platform software device-mode
Operating device-mode: Autonomous
Device-mode bootup status:
------------------------------------
Device# show platform software chasfs r0 brief | inc device_managed_mode
/tmp/chassis/local/rp/chasfs/etc/device_managed_mode : [autonomous]
/tmp/fp/chasfs/etc/device_managed_mode : [autonomous]
Device# show version | inc Last reload
Last reload reason: Enabling autonomous-mode
```

Sample configuration output for CSR1000v instance in controller mode

```
Device# show version | inc operating
Router operating mode: Controller-Managed
Device# show platform software device-mode
Operating device-mode: Controller
Device-mode bootup status:
```

```
-----------------------------------
Success
Device# show platform software chasfs r0 brief | inc device_managed_mode
/tmp/chassis/local/rp/chasfs/etc/device_managed_mode : [controller]
/tmp/fp/chasfs/etc/device_managed_mode : [controller]
Device# show version | inc Last reload
Last reload reason: Enabling controller-mode
```

Upgrading from existing IOS XE and SD-WAN Images

Non-SD-WAN Image to Autonomous Mode Upgrade

If you are an existing non-SD WAN user and you are upgrading to the 17.2 release (autonomous mode), you can directly perform the upgrade. That is, you can directly upgrade an existing instance from either UCMk9 or universalk9 to 17.2 universalk9 images.

For example, if you use the image csr1000v-universalk9.16.12.02s.SPA.bin, and upgrade to csr1000v-universalk9.17.xx.xx.SPA.bin, the instance boots up in the autonomous mode and the configuration from UniversalK9 16.xx.xx instance gets ported over to the 17.2 autonomous mode instance.

To know more about the day0/bootstrap configuration for each cloud, see *Day 0 and Custom Data Configuration for the Cloud Service Providers* in this feature document.

SD-WAN Image to Controller Mode Upgrade

If you are an existing SD-WAN user, and you upgrade to the IOS XE 17.2. release, for example, you upgrade from Csr1000v-ucmk9.16.xx.xx.SPA.bin to CSR1000v-universalk9.16.xx.xx.bin, the instance boots up in the controller mode automatically and the configuration from 16.xx.xx cedge instance gets ported over to the 17.2.1 controller mode instance.

Non-SD-WAN Image to Controller Mode Upgrade

If you are an existing non-SD-WAN user (universalk9 user) who wants to upgrade to the IOS XE 17.2 release (Controller mode), perform a mode switch. In this case, the existing configuration data is deleted. To proceed with the router configuration the controller mode, see Upgrading from Existing IOS XE and SD-WAN image to Cisco IOS XE 17.2 and Later.

Note After you install the Cisco IOS XE 17.2 image, if you want to switch to the autonomous mode, for CSR1000v instance running on public clouds, provide the appropriate bootstrap configuration. For CSR 1000v instances running on private clouds, the instance comes up with no bootstrap configuration unless you mount an ISO file with the iosxe_config.txt file or the ovf-env.xml file.

Downgrade from Cisco IOS XE 17.2 and Later

Downgrading to a fresh install of old image versions brings the device to Day 0 configuration. For example, if you have never installed Cisco IOS XE 16.12 on your device and attempt to downgrade from Cisco IOS XE Release or later releases to Cisco IOS XE 1612, the following warning displays:

Warning You are trying to activate an old image which will remove all device configuration and bring the device back to day-0 state. To proceed, use the **clean** option at activation.

For all the downgrade scenarios, see Downgrade from Cisco IOS XE 17.2 and Later section in the Cisco SD-WAN Getting Started Guide.

Switching Between Autonomous and Controller Modes

To determine the current mode of your device, run the **show version | inc operating** command. The following table lists the commands and the configuration files needed for switching between the two modes:

Current Mode	Command to Switch Mode	Mode Changes To	Configuration File and Location	Configuration Example

Autonomous mode	controller mode enable	Controller mode	ciscosdwan.cfg on bootflash, CDROM, or CDROM1
			ciscosdwan_cloud_init.cfg on bootflash, CDROM, or CDROM1
				`cisco#controller-mode enable`
				`Enabling controller mode will erase the nvram filesystem, remove all configuration files, and reload the box!`
				`Ensure the BOOT variable points to a valid image`
				`Continue? [confirm]`

Controller mode	controller mode disable	Autonomous mode	ciscortr.cfg in any file system available to the device	`cisco#controller-mode disable`
				`Disabling controller mode will erase the nvram filesystem, remove all configuration files, and reload the box!`
				`Ensure the BOOT variable points to a valid image`
				`Continue? [confirm]`

Note After you execute these commands, the device reloads before the mode change takes effect.

If you want to perform a mode switch and you have a bootstrap configuration file for your configuration, ensure that you copy the file to the bootflash.

If you're using a Cisco CSR1000V 17.3.x instance, upload the ciscosdwan_cloud_init.cfg bootstrap file with the *aaa authorization exec default local* command to the router. This command configures the AAA authorization, checks the local database, and allows you to run an EXEC shell.

To know more about uploading the bootstrap file in controller mode, see https://www.cisco.com/c/en/us/td/docs/routers/sdwan/configuration/sdwan-xe-gs-book/hardware-and-software-installation.html#c_On_Site_Bootstrap_Process_for_SD_WAN_Devices_1248xml2.

In the case of public clouds, to provide access to the Cisco CSR 1000V instance in either modes, the instance comes up with bare minimum bootstrap configuration. You can use an SSH to log in to the instance during the initial bootup stage or after performing the mode switch operation.

Frequently Asked Questions

-

Q. I have been using Cisco IOS XE image until now. Which mode should I now choose?

A. If you have been using the Cisco IOS XE universalk9 image so far, deploy the IOS XE 17.2 image and enter the autonomous mode. For more information, see *Bootstrap Configuration* section in this chapter.

Q. If I am upgrading to the CSR 1000v 17.2 release, do I need to provide the bootstrap configuration?

A. Prior to Cisco IOS XE Release 17.2, the Cisco IOS XE SD-WAN images were in the ucmk9 format while the Cisco IOS XE was in the universalk9 format. If you are an existing non-SD WAN user and are upgrading to the IOS XE 17.2 release (autonomous mode), you can directly perform the upgrade. You need not perform the Day 0 or the custom data configuration again.

For CSR 1000v instance running on Azure, the device uses the custom data that you provided the first time you configured your CSR 1000v instance.

For CSR 1000v instances running on AWS and GCP, the device fetches the custom data from the cloud service provider.

Q. What happens to my custom data configuration after switching modes?

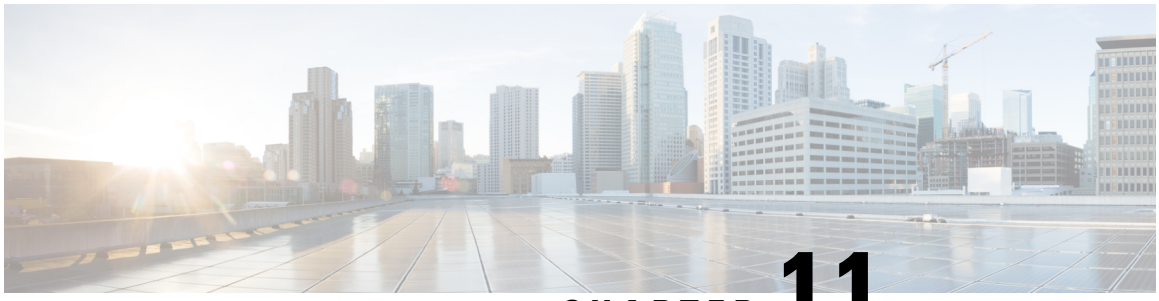
A. The existing configuration data is deleted. Perform the bootstrap or custom data configuration just as you do for a fresh installation.

Q. What happens to my custom data after a factory reset?

A. When you perform a factory reset, the configuration and the files present on the disk are erased. The router boots up like a fresh install and looks for configuration files at the appropriate location specified. This action determines the mode and the associated configuration.

Q. Can I deploy my CSR 1000v instance in the controller mode with PayG license?

A. If you use the PayG licensing model, you cannot deploy the CSR 1000v instance in the controller mode or switch to the Controller mode, as this mode does not support the PayG licensing model.



CHAPTER 11

Installing Cisco CSR 1000v Licenses

Activating Cisco CSR 1000v Licenses

When the Cisco CSR 1000v or Cisco ISRv first boots, it boots in evaluation mode. The network interfaces are activated but throughput is limited to 2.5 Mbps and the feature support is limited. Activate the software licenses to obtain the throughput and feature support provided by the license. For information about the available licenses in your software version, see the Cisco CSR 1000v Release Notes. The Cisco CSR 1000v and Cisco ISRv support the following options to activate the software licenses:

Cisco Software Licensing (CSL)	Installing the Cisco CSR 1000v/Cisco ISRv licenses using Cisco Software Licensing (CSL) uses a similar process to that of other Cisco router platforms. See Installing CSL Evaluation Licenses for Cisco IOS XE 3.13S and Later, on page 162 and subsequent sections.
Cisco Smart Licensing	Cisco CSR 1000v and Cisco ISRv support activation using Cisco Smart Licensing. (Cisco IOS XE Release 3.15S and later.) See Cisco Smart Licensing, on page 179.

Cisco Software Licensing (CSL)

Installing CSL Evaluation Licenses for Cisco IOS XE 3.13S and Later

In Cisco IOS XE 3.13S and later, including IOS XE Denali 16.2 and later, the Cisco CSR 1000v/ISRv first boots with the AX feature set enabled and the maximum throughput limited to 100 Kbps. The following evaluation licenses are available:

- AX feature set with 50 Mbps maximum throughput
- APPX feature set with 5 Gbps maximum throughput

The evaluation licenses are available for download at the Cisco Software Licensing portal.

Note If you are installing an evaluation license for a feature set with a maximum throughput of 10 Gbps, then additional configuration is required to support the 10 Gbps interface. For more information, see the *Configuring an Interface for 10Gbps Maximum Throughput* section in this guide.

Perform the following steps after the router first boots:

SUMMARY STEPS

1. **enable**
2. **show license udi**
3. Log on to the Cisco Software Licensing portal to obtain the evaluation license: http://www.cisco.com/go/license.
4. **license install** *stored-location-url*
5. **configure terminal**

6. **license boot level {ax | appx}**
7. **end**
8. **write memory**
9. **reload**
10. **show license detail**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**show license udi** **Example:** `show license udi`	Displays all the UDI values that can be licensed in a system. • You need the UDI of the device to obtain the evaluation license.
Step 3	Log on to the Cisco Software Licensing portal to obtain the evaluation license: http://www.cisco.com/go/license.	**a.** Click on **Continue to Product Registration**. **b.** Click on **Get Other Licenses** and select **Demo and Evaluation.** **c.** Under Product Family, select **Router & Switches**. **d.** Under **Product**, select **Cisco Cloud Services Router 1000v** **e.** Click **Next**. **f.** Select the evaluation license. **g.** Select whether the evaluation license will be used on an Amazon AWS instance, a standalone deployment, or other deployment. **h.** In the UDI Serial Number field, enter the 11-character UDI obtained in step 2. Note that the UDI is case-sensitive, and should be entered in all capital letters. **i.** Specify the Product ID; for example, CSR1000v. **j.** Download the evaluation license.
Step 4	**license install** *stored-location-url* **Example:**	Installs the evaluation license obtained in the previous steps.

	Command or Action	Purpose
	`license install bootflash:90NVHJ3C26E_20140724194119019.lic`	• Accept the End-User License Agreement when prompted.
Step 5	**configure terminal** **Example:** `configure terminal`	Enters global configuration mode.
Step 6	**license boot level {ax | appx}** **Example:** `license boot level ax`	Activates the evaluation license on the router upon the next reload. Select **ax** if installing the AX feature set evaluation license. Select **appx** if installing the APPX feature set evaluation license. Accept the end user license agreement when it is prompted.
Step 7	**end** **Example:** `end`	Exits global configuration mode.
Step 8	**write memory** **Example:** `write memory`	Saves the running configuration to NVRAM.
Step 9	**reload** **Example:** `reload`	Restarts the router to enable the feature set and the maximum throughput supported by the evaluation license. The router reloads with the evaluation license activated. The evaluation license expires 60 days from the time it is activated.
Step 10	**show license detail** **Example:** `show license detail`	Displays the license information.

What to do next

Note If you are installing an evaluation license for a feature set with a maximum throughput of 10 Gbps, then additional configuration is required to support the 10 Gbps interface. For more information, see the *Configuring an Interface for 10Gbps Maximum Throughput* section in this guide.

Installing CSL Regular Licenses for Cisco IOS XE 3.13S and Later

In Cisco IOS XE 3.13S and later, including IOS XE Denali 16.3 and later, the Cisco CSR 1000v/ISRv first boots in limited mode with the AX feature set enabled and the maximum throughput limited to 100 Kbps. You can generate multiple licenses for the router from one PAK. The purchased PAK determines the number of licenses you can generate.

Repeat these steps for each license available for your PAK.

Note If you installed a license that supports a maximum throughput of 10 Gbps, then additional configuration is required to support the 10 Gbps interface. For more information, see the *Configuring an Interface for 10Gbps Maximum Throughput* section in this guide.

SUMMARY STEPS

1. Obtain the PAK.
2. **enable**
3. **show license udi**
4. Convert the PAK to a license by entering the PAK and the UDI into the Cisco Product License registration portal.
5. **license install** *stored-location-url*
6. **configure terminal**
7. **license boot level {ipbase | security** | **ax | appx**}
8. **end**
9. **write memory**
10. **reload**
11. **show license detail**
12. **platform hardware throughput level MB**{**10 | 100 | 1000 | 10000 | 250 | 2500 | 50 | 500 | 5000**}

DETAILED STEPS

Procedure

Step 1 Obtain the PAK.

The PAK is provided to you when you order or purchase the right to use a feature set.

- The PAK serves as a receipt and is used as part of the process to obtain a license.

Step 2 **enable**

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 3 **show license udi**

Displays all the UDI values that can be licensed in a system.

- You need the UDI of the device as part of the process to obtain a license.

Step 4 Convert the PAK to a license by entering the PAK and the UDI into the Cisco Product License registration portal.

Example:

When entering the UDI, enter only the 11-character serial number; for example,

```
966975BITWG
```

. The UDI is case-sensitive, and should be entered in all capital letters.

After entering the appropriate information, you will receive an e-mail containing the license information that you can use to install the license:

- Copy the license file received from the Cisco Product License Registration portal to the appropriate file system on the device.

Step 5 **license install** *stored-location-url*

Example:

```
Router# license install bootflash:90NVHJ3C26E_20140724194119019.lic
```

Installs the license.

- Accept the end-user license agreement if prompted.

Step 6 **configure terminal**

Example:

```
Router# configure terminal
```

Enters global configuration mode.

Step 7 **license boot level {ipbase | security | ax | appx}**

Example:

```
Router(config)# license boot level ax
```

Activates the license on the router upon the next reload.

Step 8 **end**

Example:

```
Router(config)# end
```

Exits configuration mode.

Step 9 **write memory**

Example:

```
Router# write memory
```

Saves the running configuration to NVRAM.

Step 10 **reload**

Example:

```
Router# reload
```

Restarts the router to enable the feature set and the maximum throughput supported by the license.

Note: If you are installing an AX license, you do not need to restart the router.

Step 11 **show license detail**

Example:

The following is an example of the **show license detail** command showing an installed active license:

```
Router# show license detail
Index: 1        Feature: sec_100M                         Version: 1.0
        License Type: Permanent
        License State: Active, In Use
        License Count: Non-Counted
        License Priority: Medium
        Store Index: 0
        Store Name: Primary License Storage
```

Displays the license information.

Step 12 **platform hardware throughput level MB**{**10 | 100 | 1000 | 10000 | 250 | 2500 | 50 | 500 | 5000**}

Example:

```
Router(config)# platform hardware throughput level 500
```

(Optional) Changes the maximum throughput level.

Note: After issuing this command, you do not need to restart the router.

What to do next

Repeat these steps for each license available for your PAK.

Configuring an Interface for 10 Gbps Maximum Throughput

If you installed a license with maximum throughput with 10 Gbps, then additional configuration is required to obtain the 10 Gbps throughput on an interface. Perform the following additional steps.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface GigabitEthernet** *number*
4. **no negotiation auto**
5. **speed 10000**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface GigabitEthernet** *number* **Example:** `Router(config)# interface GigabitEthernet1`	Enters interface configuration mode.
Step 4	**no negotiation auto** **Example:** `Router(config-if)# no negotiation auto`	Disables the autonegotation protocol setting on the interface.
Step 5	**speed 10000** **Example:** `Router(config-if)# speed 10000`	Configures the interface speed to 10 Gbps.

Installing CSL Feature Add-on Licenses for Cisco IOS XE 3.13S and Later

Understanding the Cisco CSR 1000v Memory Allocation

You can use feature add-on licenses to add memory to the Cisco CSR 1000v/Cisco ISRv. Memory is allocated to both the IOSd component and the data plane component. The amount of the memory allocation is dependent on the licenses installed.

You can install multiple 4 GB add-on licenses. You can add 4 GB of additional memory by installing the broadband feature license and then install further 4 GB add-on licenses.

The following table lists how the memory is allocated depending on the amount of VM Memory and the feature licenses installed.

Note Restrictions apply when installing memory add-on licenses with a broadband feature license. For more information, see the I*nformation About Installing Broadband Feature License* and the *Installing Broadband Feature License* sections in this guide.

Note The Cisco CSR 1000v is no longer available with a VM Memory of either 2.5 GB or 6 GB.

Table 31: Cisco CSR 1000v Memory Allocation with Memory Add-on Licenses

VM Memory	Default Memory Allocation	One 4 GB add-on license or one broadband license	(Two 4 GB add-on licenses) or (one broadband license + one 4 GB add-on license)	(Three 4 GB add-on licenses) or (one broadband license + two 4 GB add-on licenses)
4 GB (Additional memory allocation using an add-on license or broadband license is not available for this level of VM memory.)	IOSd = 2.5 GBDataplane = 1.5 GB	NA	NA	NA
8 GB	IOSd = 2.5 GBDataplane = 1.5 GB	IOSd = 5.5G Dataplane = 2.5G	NA	NA
12 GB	IOSd = 2.5 GBDataplane = 1.5 GB	IOSd = 5.5G Dataplane = 2.5G	IOSd = 9.5G Dataplane = 2.5G	NA
16 GB	IOSd = 2.5 GBDataplane = 1.5 GB	IOSd = 5.5G Dataplane = 2.5G	IOSd = 9.5G Dataplane = 2.5G	IOSd = 13.5G Dataplane = 2.5G

Further Information about Memory Add-on Licenses

This section seeks to explain some misleading memory usage values that may be shown after installing add-on licenses. Installing add-on memory provides additional memory that is assigned to the main IOS-XE process (IOSd). For example, if you add three 4 GB add-on licenses you may gain approximately 11 GB memory. However, bear in mind that adding memory may not solve underlying issues with your configuration and the additional memory may not be necessary.

If you add two or three memory add-on licenses, you may see misleading messages such as the following error log message:

```
%PLATFORM-3-ELEMENT_CRITICAL: R0/0: smand: RP/0: Used Memory value 96% exceeds critical
level 93%
```

A similar high usage value is displayed by a **show platform** command such as **show platform software status control-processor brief**. See Example 1 below.

Example 1

In this example a Cisco CSR 1000v running Cisco IOS XE 16.6.2 has 2 x 4 GB memory add-on licenses. The displayed information indicates a critically high memory usage.

```
# show platform software status control-processor brief
...
Memory (kB)
 Slot  Status    Total     Used (Pct)     Free (Pct) Committed (Pct)
  RP0 Critical 12242316 11775260 (96%)   467056 ( 4%)  12255384 (100%)      << 96% Critical
```

If you were able to have access to the underlying Linux system you could find that only less memory than 96% is being used. For example, internally the following Linux command shows only 81% usage—based on used memory as a percentage of total memory.

```
free -m
              total        used        free      shared   buff/cache   available
Mem:          11955        9708          76         758        2169        1383
```

If you were then to add a third 4 GB add-on license, making a total of three add-on licenses, the 96% memory usage that is displayed by the **show platform** command would not be significantly reduced.

If you use the `show processes memory sorted` command, as shown in Example 2 below, you get a better indication of the memory usage.

Example 2

This example shows the difference between using a Cisco CSR 1000v with no add-on licenses, and a Cisco CSR 1000v with two add-on licenses.

1. CSR 1000v with no add-on licenses.

   ```
   # show processes memory sorted
   ...
   Processor Pool Total: 2458193040 Used:  239241616 Free: 2218951424    << 239 MB used
   ```

2. CSR 1000v with two add-on licenses.

   ```
   # show processes memory sorted
   ...
   Processor Pool Total: 9625210000 Used: 1231337528 Free: 8393872472    <<  1.2 GB used
   ```

This shows that even considering the additional 700 MB extra processing needs, the memory that is being used is quite low. Therefore, using two add-on licenses for this processing requirement may be unnecessary.

Installing Memory Add-on License

Beginning with Cisco IOS XE 3.13S, you can add memory in 4 GB increments to enable control plane scaling using the memory add-on license (L-CSR-MEM-4G=). The following prerequisites apply:

- The base feature license must be installed.
- The VM must have enough memory allocated to accommodate the additional memory. For more information, see the table in the section *Understanding the Cisco CSR 1000v Memory Allocation* in this guide.

SUMMARY STEPS

1. Obtain the PAK.
2. **enable**
3. **show license udi**
4. Convert the PAK to a license by entering the PAK and the UDI into the Cisco Product License registration portal: http://www.cisco.com/go/license
5. **show platform software vmemory info**
6. **configure terminal**
7. **platform memory add** *memory*
8. **end**
9. **license install** *stored-location-url*
10. **write memory**
11. **reload**
12. **show license detail**
13. **show platform software vmemory info**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	Obtain the PAK.	The PAK is provided to you when you order or purchase the right to use a feature set. • The PAK serves as a receipt and is used as part of the process to obtain a license.
Step 2	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 3	**show license udi** **Example:** `Router# show license udi`	Displays all the UDI values that can be licensed in a system. • You need the UDI of the device as part of the process to obtain a license.
Step 4	Convert the PAK to a license by entering the PAK and the UDI into the Cisco Product License registration portal: http://www.cisco.com/go/license	After entering the appropriate information, you will receive an e-mail containing the license information that you can use to install the license: • Copy the license file received from the Cisco Product License Registration portal to the appropriate file system on the device.
Step 5	**show platform software vmemory info**	Verifies the current memory allocation on the Cisco CSR 1000v/Cisco ISRv. The display shows the memory upgrade

	Command or Action	Purpose
		license limit, indicating the maximum amount of additional memory you can add.
Step 6	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 7	**platform memory add** *memory* **Example:** `Router(config)# platform memory add 4096`	Adds the memory allocation to the router to accommodate added memory license(s). Add 4096 MB for each memory license you are planning to install. For example, if you plan to add three memory licenses, you would add 12288 MB of memory.
Step 8	**end** **Example:** `Router(config)# end`	Exits configuration mode.

Step 9	**license install** *stored-location-url*
	Example:
	`Router# license install bootflash:90NVHJ3C26E_20140724194119019.lic`
	Example:
	`4096 MB memory has been added to the system`
	Example:
	`Execute 'write memory' to persist this change`
		Installs the memory add-on license.

Step 10	**write memory** **Example:** `Router# write memory`	Saves the running configuration to NVRAM.
Step 11	**reload** **Example:** `Router# reload`	Restarts the router to enable the memory add-on license to be activated.
Step 12	**show license detail** **Example:** `Router# show license detail`	Displays the license information to verify the installation of the memory license(s).

	Command or Action	Purpose

Step 13	**show platform software vmemory info**
	Example:
	
	`Router# show platform software vmemory info`
	`Memory Upgrade Limits:`
	`Total System Memory:3894 MB`
	`Memory From Upgrade Licenses:N/A(Smart License Enabled)`
	`Memory From Feature Licenses:N/A(Smart License Enabled)`
	`Memory Available For Upgrade:`
	`Available System Memory:0 MB`
	`Available Upgrade Licensed Memory:N/A(Smart License Enabled)`
	`Available Feature Licensed Memory:N/A(Smart License Enabled)`
	`Current Memory Allocation:`
	`IOSD:2358 MB (default) + 0 MB upgrade`
	`Data Plane:1536 MB (default) + 0 MB upgrade`
		Verifies the updated memory allocation on the router.

Example

The following is an example of the **show license** command with details of a memory add-on license shown:

```
Router# show license
Index 1 Feature: ax
Index 2 Feature: mem_4G
        Period left: Life time
        License Type: Permanent
        License State: Active, In Use
        License Count: 1/1/0  (Active/In-use/Violation)
        License Priority: Medium
```

Information About Installing Broadband Feature License

The Cisco CSR 1000v/ Cisco ISR v support the Broadband Network Gateway feature set and the Intelligent Services Gateway feature set. The required broadband feature license (For the Cisco CSR 1000v: L-CSR-BB-1K=) provides up to 4 GB of additional memory and support for up to 1000 broadband sessions.

The following restrictions apply:

- The APPX feature license with a minimum of 1 Gbps maximum throughput must be installed.
- You can install multiple broadband feature licenses to increase the number of broadband sessions. However, installing additional broadband feature licenses will not add more memory. To add more memory beyond the 4 GB installed with the first broadband feature license, you must install a separate memory add-on license.
- If both a broadband feature license and memory add-on licenses are installed, then the broadband license takes higher priority than any memory add-on licenses installed. When the Cisco CSR 1000v/ Cisco ISRv is reloaded, the broadband feature license takes effect first, before any installed memory add-on licenses.

- We recommend that you install the broadband feature license before installing any memory add-on licenses.
- The VM must have enough memory allocated to accommodate the additional memory. See Understanding the Cisco CSR 1000v Memory Allocation, on page 168 for more information.

For more information about configuring broadband support, see Broadband Access Aggregation and DSL Configuration Guide and Intelligent Services Gateway Configuration Guide.

Installing Broadband Feature License

SUMMARY STEPS

1. Obtain the PAK.
2. **enable**
3. **show license udi**
4. Convert the PAK to a license by entering the PAK and the UDI into the Cisco Product License registration portal http://www.cisco.com/go/license
5. **show platform software vmemory info**
6. **configure terminal**
7. **platform broadband** {**1K** | **2K** | **3K** | **4K** | **5K** | **6K** | **7K** | **8K**}
8. **platform memory add** *memory*
9. **end**
10. license install *stored-location-url*
11. **write memory**
12. **reload**
13. **show license detail**
14. **show platform software vmemory info**
15. (Optional) Install memory add-on licenses as needed. For more information, see the *Installing a Memory Add-on License* section in this guide.

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	Obtain the PAK.	The PAK is provided to you when you order or purchase the right to use a feature set. • The PAK serves as a receipt and is used as part of the process to obtain a license.
Step 2	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 3	**show license udi** **Example:** `Router# show license udi`	Displays all the UDI values that can be licensed in a system. • You need the UDI of the device as part of the process to obtain a license.
Step 4	Convert the PAK to a license by entering the PAK and the UDI into the Cisco Product License registration portal http://www.cisco.com/go/license	After entering the appropriate information, you will receive an e-mail containing the license information that you can use to install the license: • Copy the license file received from the Cisco Product License Registration portal to the appropriate file system on the device.
Step 5	**show platform software vmemory info**	Verifies the current memory allocation on the router. The display shows the memory upgrade license limit, indicating the maximum amount of additional memory you can add.
Step 6	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 7	**platform broadband** {**1K** | **2K** | **3K** | **4K** | **5K** | **6K** | **7K** | **8K**} **Example:** `Router(config)# platform broadband 1K`	Adds support for the number of broadband sessions to accommodate the added broadband feature license(s). You can add 1000 sessions for each broadband feature license you are planning to install. For example, if you plan to add two broadband feature licenses, enter the value as 2K.
Step 8	**platform memory add** *memory* **Example:** `Router(config)# platform memory add 4096`	(Optional) Adds the memory allocation to the router to accommodate added memory license(s). Add 4096 MB for each memory license you are planning to install. For example, if you plan to add two memory licenses, add 8192 MB of memory. **Note** The broadband feature license adds 4 MB of additional memory. If you want to add more memory, you must use this command. Adding more broadband feature licenses does not add more memory.
Step 9	**end** **Example:** `Router(config)# end`	Exits configuration mode.
Step 10	license install *stored-location-url* **Example:** `Router# license install bootflash:90NVHJ3C26E_20140724194119019.lic`	Installs the broadband feature license and any additional memory add-on licenses.

	Command or Action	Purpose

	Example:
	`bootflash:90NVHJ3C26E_20140724194119019.lic`
	Example:
	`4096 MB memory has been added to the system`
	Example:
	`Execute 'write memory' to persist this change`	

Step 11	**write memory** **Example:** `Router# write memory`	Saves the running configuration to NVRAM.
Step 12	**reload** **Example:** `Router# reload`	Restarts the router to enable the memory add-on license to be activated.
Step 13	**show license detail** **Example:** `Router# show license detail`	Displays the license information to verify the installation of the broadband feature license(s) and memory license(s).
Step 14	**show platform software vmemory info**	Verifies the updated memory allocation on the router.
Step 15	(Optional) Install memory add-on licenses as needed. For more information, see the *Installing a Memory Add-on License* section in this guide.	

Example

The following is an example of the **show license** command showing details of a broadband feature license:

```
# show license | begin bb
Index 76 Feature: bb_1K
Period left: Life time
License Type: Permanent
License State: Active, In Use
License Count: 1/1/0  (Active/In-use/Violation)
License Priority: Medium
Index 77 Feature: mem_4G
```

Troubleshooting CSL License Issues

Determining the License Status

You can install multiple licenses on a Cisco CSR 1000v/ ISRv. To determine if a license is active, enter the **show license** or **show license detail** command. The display indicates the license status. The following are the possible states for the license:

- Active, In Use

  This state indicates that the license is active and is in use by the Cisco CSR 1000v.

- Active, Not in Use

  This state indicates that the license is installed on the Cisco CSR 1000v, but is not currently being used.

- Inactive

  This state indicates that the license is installed on the Cisco CSR 1000v but is no longer valid. For example, a license that has reached the end of the subscription term is shown as inactive.

The following example shows that a Cisco CSR 1000v has two licenses installed: an AX technology license and a Security technology license:

```
router# show license detail

Index: 1        Feature: ax_1G                          Version: 1.0        License Type:
 Paid Subscription        Start Date:         N/A, End Date: Nov 10 2014        License
State: Active, In Use        License Count: Non-Counted        License Priority: Medium
     Store Index: 0        Store Name: Primary License StorageIndex: 2        Feature:
sec_1G                          Version: 1.0        License Type: Permanent        License
 State: Active, Not in Use        License Count: Non-Counted        License Priority: Medium
        Store Index: 1        Store Name: Primary License Storage
```

The AX technology license is shown as Active and in use, while the Security technology license is Active but not in use. To use the Security technology license, the **license boot level** command needs to be configured to “security” and the Cisco CSR 1000v must then be reloaded.

The following example of the **show version** command shows that the Cisco CSR 1000v has an AX technology license installed, but that the license boot level command has been set to “security”, but the Cisco CSR 1000v has not yet been reloaded.

```
router# show version | inc Level

  License Level: ax   Next reload license Level: security
```

Migrating Technology Package Licenses to Cisco IOS XE 3.13S

Starting with Cisco IOS XE 3.13S, the names of the technology package licenses changed as shown below.

- The Standard technology package was changed to the IPBase technology package.
- The Advanced technology package was changed to the Security technology package.

- The Premium technology package was changed to the AX package.

The base feature content for each license is the same as previously, but the names as shown in the licenses and display output have changed. If you migrated either a Standard or Advanced technology package license from a previous version to Cisco IOS XE 3.13S, then the show version and show license commands display the old license names, which is expected behavior. The new license names display when you enter the **show running configuration** command.

In the following example, the **show running configuration** command following the migration shows the new "security" technology package :

```
Router# show running | include level
license boot level security
```

However, in the **show version** output, the migrated license displays as the old "advanced" technology package name, as shown in the following example:

```
Router# show version | include License Level
License Level: advanced
```

In the **show license detail** output, the feature license also shows the old advanced license package name, as shown in the following example:

```
Router# show license detail
Index: 1        Feature: adv_100M                         Version: 1.0
        License Type: Permanent
        License State: Active, In Use
        License Count: Non-Counted
        License Priority: Medium
        Store Index: 0
        Store Name: Primary License Storage
```

No further configuration is required. To verify the correct feature set name for the migrated license, use the **show running configuration** command.

Determining the AWS License Type

When you deploy a Cisco CSR 1000v instance from a Cisco CSR 1000v Amazon Machine Image (AMI), the license that is displayed differs depending on whether you deployed a Bring Your Own License (BYOL) or an hourly-usage license.

- If the **show license** command shows the license as "advance, internal_service" or a similar designation, then the instance uses an hourly-usage license purchased on Amazon Web Services.

The following example displays the license information for an hourly-usage instance:

```
router# show license
Index 1 Feature: ax           Index 2 Feature: internal_service
```

- If the **show license** command shows a list of supported licenses with various throughput levels, then the instance is a BYOL instance.

The following example displays the license information for a BYOL instance:

```
router# show license
```

```
Index 1 Feature: advanced
Index 2 Feature: standard
Index 3 Feature: ax
Index 4 Feature: security
Index 5 Feature: lite
Index 6 Feature: appx
Index 7 Feature: ipbase
Index 8 Feature: prem_10M
Index 9 Feature: prem_50M
Index 10 Feature: prem_100M
Index 11 Feature: prem_250M
Index 12 Feature: prem_500M
Index 13 Feature: prem_500M_8G
Index 14 Feature: prem_1G
Index 15 Feature: prem_1G_16G
Index 16 Feature: prem_2500M
Index 17 Feature: prem_5G
Index 18 Feature: prem_10G
Index 19 Feature: prem_200G
Index 20 Feature: ax_10M
Index 21 Feature: ax_50M
Index 22 Feature: ax_100M
Index 23 Feature: ax_250M
Index 24 Feature: ax_500M
Index 25 Feature: ax_500M_8G
Index 26 Feature: ax_1G
```

- The **license boot level** and **platform hardware throughput-level** commands are not available with hourly-usage license. These commands are only supported on Cisco CSR 1000v instances with BYOL licenses.

Cisco Smart Licensing

Note If you are using CSR1000v release 16.10.1 or later, for Smart Licensing information, refer to the Smart Licensing Guide for Access and Edge Routers.

Beginning with Cisco IOS XE Release 3.15S, the Cisco CSR 1000v/ Cisco ISRv support activation using Cisco Smart Licensing.

- To use Cisco Smart Licensing, you must first configure the Call Home feature and obtain Cisco Smart Call Home Services.
- For Cisco IOS XE 3.15S and later, and IOS XE Denali 16.3 and later, the following Cisco IOS XE technology packages are supported: IPBase, Security, AX and APPX
- Cisco Smart Licensing uses the Cisco Smart Software Manager for managing licenses. To access the Cisco Smart Software Manager, use the following URL: https://software.cisco.com/#module/SmartLicensing

For more information about Cisco Smart Software Manager, see the Cisco Smart Software Manager User Guide , which is accessible from the Cisco Smart Software Manager tool.

Prerequisites for Cisco Smart Licensing

Before enabling Cisco Smart Licensing on the router, Cisco Smart Call Home must be configured by following the steps in Configuring Call Home for Smart Licensing, on page 180.

Note For further information on Smart Call Home, see Obtaining Smart Call Home Services, on page 272 and Configuring and Enabling Smart Call Home, on page 275.

Restrictions for Cisco Smart Licensing

- You cannot use the Web UI to configure SLP related commands for Cisco CSR1000V or Cisco ISRv 17.3.x and earlier versions.

Configuring Call Home for Smart Licensing

Describes how to configure and activate Call Home specifically for Smart Licensing. This is a prerequisite for configuring Smart Licensing on the Cisco CSR 1000v/ ISRv.

Note For more information in general about configuring Call Home, see Configuring Call Home for the Cisco CSR 1000v, on page 271.

Procedure

Step 1 **configure terminal**

Example:

```
Router# configure terminal
```

Enters configuration mode.

Step 2 **service call-home**

Example:

```
Router(config)# call-home
```

Activates the call-home feature.

Step 3 **call-home**

Example:

```
Router(config)# call-home
```

Enters the Call Home configuration submode.

Step 4 **profile** *name*

Example:

```
Router(config-call-home)# profile
 CiscoTAC-1
```

Enters the Call Home destination profile configuration submode for the specified destination profile. If the specified destination profile does not exist, it is created.

Step 5 **destination transport-method http**

Example:

```
Router(cfg-call-home-profile)# destination transport-method email
```

Enables the HTTP message transport method.

Step 6 **no destination transport-method email**

Example:

```
Router(cfg-call-home-profile)# no destination transport-method email
```

Disables email as the transport method.

Step 7 **destination address http** *url*

url = https://tools.cisco.com/its/service/oddce/services/DDCEService —address of the Cisco Smart Call Home Server.

Example:

```
Router(cfg-call-home-profile)# destination address email
https://tools.cisco.com/its/service/oddce/services/DDCEService
```

Configures the destination email address or URL to which Call Home messages are sent.

Note
When entering a destination URL, include either **https://** or **http://** , depending on whether the server is a secure server, or not.

Enabling Cisco Smart Licensing

To enable Cisco Smart Licensing and register your device, perform the following steps:

Procedure

Step 1 Execute the **configure terminal** command.

Example:

```
Router# configure terminal
```

Enters the global configuration mode.

Step 2 Execute the **license smart enable** command.

Example:

```
Router(config)# license smart enable
```

This command enables Cisco Smart Licensing and disables Cisco Software Licensing (CSL).

Step 3 To further establish connectivity, perform the following optional steps:

a) Execute the **ip http client source-interface <interface>** command.
b) Execute the **ip domain lookup source-interface <interface>** command.
c) Execute the **ip name-server vrf mgmt <ip address>** command.

Step 4 **exit**

Example:

```
Router(config)# exit
```

Exits the configuration mode.

What to do next

After you enable the Cisco Smart Licensing, the Cisco CSR 1000v instance is no longer in the evaluation mode. The technology level and the throughput level supported by your license takes effect. For more information about managing the technology package and throughput license attributes, see Understanding the License-Based Restriction on Aggregate Bandwidth, on page 217 and Managing Throughput Notifications, on page 219.

Use the **show running-config** command to verify whether the Cisco Smart Call Home is enabled. The following configuration should be included:

```
call-home
  profile "CiscoTAC-1"
  active
  destination transport-method http
  no destination transport-method email
  destination address http https://tools.cisco.com/its/service/oddce/services/DDCEService
```

Verify that the **destination address** command points to the URL of the Cisco Smart Software Agent as shown above. If the URL is not configured, you must manually configure the **destination address http** command to point to the URL.

After the connectivity is established, register the router with the Cisco Licensing Cloud. For example, see the Registering the Router with the Cisco Licensing Cloud, on page 201 section.

Smart Licensing System Messages

This section lists the smart licensing system messages for the Cisco CSR 1000v and Cisco ISRv. The more severe system messages are listed first. For more information on system messages, see System Message Logging. For more information on system logging commands, see the Cisco IOS Configuration Fundamentals Command Reference.

%SMART_LIC-2-PLATFORM_ERROR

Message:

```
Smart Licensing has encountered an internal software error. Contact TAC: %s
```

Explanation:

Smart Licensing Agent has encountered an internal problem with the platform.

Recommended Action:

Contact Cisco TAC.

%SMART_LIC-2-XDM_DISPATCH_LOOP_FAILURE

Message:

```
SmartAgent Admin Init Failed due to failure at XdmDispatchLoop in creating handle
```

Explanation:

This is an internal error that occurred during scheduler initialization, when trying to create an xdm handle.

Recommended Action:

Contact Cisco TAC.

%SMART_LIC-3-APPHA_DUPLICATED_INSTANCE

Message:

```
The Application, is trying set HA information for a duplicate instance.
```

Explanation:

The application is attempting to set the HA information for an entitlement instance (handle) when another duplicate instance already exists (with same entitlement tag, appHaName and appHaInstanceID).

%SMART_LIC-3-APPHA_DUPLICATED_INSTANCE

Message:

```
The Application, is trying set HA information for a duplicate instance.
```

Explanation:

The application is attempting to set the HA information for an entitlement instance (handle) when another duplicate instance already exists (with same entitlement tag, appHaName and appHaInstanceID).

%SMART_LIC-3-PLR_CONFIG_OUT_OF_SYNC

Message:

```
Trusted Store PLR Enable flag not in sync with System Configuration, TS \%s Config \%s
```

Explanation:

The Smart Licensing configuration does not match the value of the PLR enable flag in Trusted Store. This can happen if a configuration is copied onto the system and a reload occurs. If the new configuration does not contain the Smart Licensing Enable command, the value in Trusted Store does not match.

Recommended Action:

Apply the desired Smart Licensing PLR Configuration Command and persist the configuration.

%SMART_LIC-3-NOT_AUTHORIZED

Message:

```
The entitlement \%s in Not Authorized to be used. Reason: \%s
```

Explanation:

You are using a license without authorization.

Recommended Action:

Go to the Smart Licensing portal to view your entitlements and attempt to find out why you are not authorized to use this license.

%SMART_LIC-3-CONFIG_NOT_SAVED_TSCLEAR

Message:

```
The smart agent for Licensing will now be disabled because the config was not saved before
the reload
```

Explanation:

During Smart Agent initialization, if the Smart Agent state is registered and the config with the smart license enabled flag was saved before the reboot, then the configuration was not saved before the reload.

Recommended Action:

Save the configuration before reloading.

%SMART_LIC-3-AUTH_RENEW_FAILED

Message:

```
Authorization renewal with the Cisco Smart Software Manager or satellite : \%s
```

Explanation:

The Authorization renew request failed. An automatic retry occurs.

Recommended Action:

Please verify your Call Home setting and that the device has connectivity to the Cisco Smart Software Manager or satellite.

%SMART_LIC-3-AGENT_DEREG_FAILED

Message:

```
Smart Agent for Licensing DeRegistration with Cisco Smart Software Manager or satellite
failed: \%s
```

Explanation:

Smart Licensing De-registration failed. This may have been caused due to a network connection failure to the Cisco Smart Software Manager or satellite. The local registration information on the device has been removed. The registration information on the Cisco Smart Software Manager or satellitehas not been removed

Recommended Action:

Please verify your Call Home setting and that the device has connectivity to the Cisco Smart Software Manager or satellite

%SMART_LIC-3-AGENT_REG_FAILED

Message:

```
Smart Agent for Licensing Registration with the Cisco Smart Software Manager or satellite
failed: %s
```

Explanation:

Smart Licensing registration failed. Examine the included error string for a more detailed reason for the failure. This could be due to an invalid ID token or if the device is already registered.

Recommended Action:

If the ID token is invalid, it may have expired. Another reason is that you may be using an ID token from Cisco Smart Software Manager but you are registering with a CSSM satellite. If the device is already registered you may use the force option to force the registration with a new ID token. Please verify your Call Home settings and that the device has connectivity to the Cisco Smart Software Manager or CSSM satellite.

%SMART_LIC-3-AGENT_DEREG_FAILED

Message:

```
Smart Agent for Licensing DeRegistration with Cisco Smart Software Manager or satellite
failed: %s
```

Explanation:

Smart Licensing De-registration failed. This may have been caused by a network connection failure to the Cisco Smart Software Manager or satellite. The local registration information on the device has been removed. The registration information on the Cisco Smart Software Manager or satellite has not been removed.

Recommended Action:

Please verify your Call Home settings and that the device has connectivity to the Cisco Smart Software Manager or satellite.

%SMART_LIC-3-CONVERT_LIC_FAIL

Message:

```
\%s Failed to convert \%s: \%s
```

%SMART_LIC-3-UTILITY_REPORT_FAILED

Message:

```
Smart Agent for Licensing Utility has failed to send usage Report
```

%SMART_LIC-3-EVAL_EXPIRED

Message:

```
Evaluation period expired
```

Explanation:

Your evaluation period has expired. Some features may have restricted usage.

Recommended Action:

You must obtain a new ID token from the Cisco Smart Software Manager or satellite and register the device.

%SMART_LIC-3-OUT_OF_COMPLIANCE

Message:

```
One or more entitlements are out of compliance
```

Explanation:

The customer is using a license that they have not purchased or they are using more licenses than they have purchased.

Recommended Action:

You can go to the Smart Licensing portal and view your entitlements, to try and find out why the entitlements are out of compliance.

%SMART_LIC-3-INVALID_ROLE_STATE

Message:

```
The current role is not allowed to move to the new role: Current \%s New \%s
```

Explanation:

From the last role event, we can only move to certain roles. The device has moved to a role which the Smart Agent cannot follow.

Recommended Action:

Report this problem to Cisco

%SMART_LIC-3-DEPRECATED_API

Message:

```
The Deprecated function \%s has been called. This call should be replaced by \%s
```

Explanation:

This error indicates the Cisco platform team is using deprecated API functions. The platform code is calling a deprecated function. The code needs to be changed to call the new function.

Recommended Action:

Contact Cisco TAC.

%SMART_LIC-3-BAD_MODE

Message:

```
An unknown mode was specified: \%d
```

Explanation:

An invalid entitlement enforcement mode was received by the smart agent in the process of logging a syslog message. This is an internal error and should be reported to Cisco.

Recommended Action:

This is a Smart Licensing internal error. Please report this to Cisco TAC.

%SMART_LIC-3-UTILITY_EXPIRED

Message:

```
Smart Agent for Licensing Utility certificate has expired
```

Explanation:

Smart Agent for Licensing utility certificate has expired.

%SMART_LIC-3-UTILITY_RENEW_FAILED

Message:

```
Smart Agent for Licensing Utility certificate renewal failed
```

Explanation:

Smart Agent for Licensing Utility cert renew failed, this will occur once per day until the renewal is successful or the current certificate expires.

%SMART_LIC-3-INVALID_TAG

Message:

```
The entitlement tag is invalid: \%s
```

Explanation:

The entitlement tag for a license is not defined in the Cisco Smart Software Manager. This is a Cisco internal problem and should be reported to Cisco.

Recommended Action:

Report this error to Cisco

%SMART_LIC-3-BAD_NOTIF

Message:

```
A bad notification type was specified: \%d
```

Explanation:

This is a Cisco internal error. Report it to Cisco TAC.

Recommended Action:

Report this error to Cisco TAC.

%SMART_LIC-3-AGENT_REG_FAILED

Message:

```
Smart Agent for Licensing Registration with the Cisco Smart Software Manager or satellite
failed: \%s
```

Explanation:

Smart Licensing registration failed. The included error string should give a more detailed reason for the failure. This may have been due to an invalid ID token or because the device is already registered

Recommended Action:

If the ID token was invalid it may have expired or you may be using an ID token from the Smart Software Manager and you are registering with a satellite. If the device is already registered you can use the force option to force the registration with a new ID token. Please verify your Call Home setting and that the device has connectivity to the Cisco Smart Software Manager or satellite.

%SMART_LIC-3-ID_CERT_EXPIRED

Message:

```
Registration period has expired. Smart Licensing will transition to the unregistered state.
Please re-register this product to correct the problem.
```

Explanation:

The current time is outside the valid registration period in the ID certificate. This could be caused by a change in the system clock or multiple communications failures with the Cisco Smart Software Manager or satellite.

Recommended Action:

Please check the Smart Call Home settings and network connectivity to the Cisco Smart Software Manager or satellite. Also verify that your system clock is correct.

%SMART_LIC-3-ID_CERT_EXPIRED_WARNING

Message:

```
This device's registration will expire in \%s.
```

Explanation:

The registration for this device will expire at the specified time. This usually indicates a communications failure with the Cisco licensing authority.

Recommended Action:

Please verify your Call Home settings and that the device has connectivity to the Cisco Smart Software Manager or satellite.

%SMART_LIC-3-APPHA_DUPLICATED_PEER

Message:

```
The Application HA Cluster already have a member with given identity. Use the show license
usage command to see more details.
```

Explanation:

When setting up peer informations for an entitlement that supports attribute, the given peer information already exists. One of the devices may not be configured correctly or that the logic that is supposed to remove peer information is not working correctly.

%SMART_LIC-3-RESERVE_HA_FAILURE

Message:

```
The license reservation information on the active and standby does not match. Licensing HA
will not work properly: \%s
```

Explanation:

The license reservation configuration is not the same on both the active and standby. If the standby takes over as active, you will not have the same licenses available and your device may not work properly.

Recommended Action:

Change the reservation configuration in either of the nodes or both of the nodes so that they match each other.

%SMART_LIC-3-CONFIG_OUT_OF_SYNC

Message:

```
Trusted Store Enable flag not in sync with System Configuration, TS \%s Config \%s
```

Explanation:

The Smart Licensing configuration does not match the value of the enable flag in Trusted Store. This can happen if a configuration is copied onto the system and a reload occurs. If the new configuration does not contain the Smart Licensing Enable command, the value in Trusted Store does not match.

Recommended Action:

Apply the desired Smart Licensing Configuration Command and persist the configuration.

%SMART_LIC-3-REG_EXPIRED_CLOCK_CHANGE

Message:

```
Smart Licensing registration has expired because the system time was changed outside the
validity period of the registration period. The agent will transition to the un-registered
state in 60 minutes.
```

Explanation:

The system clock has been changed so that it is now outside the valid registration period. If the clock is reset to a value inside the registration validity period of 1 hour, smart licensing continues to function normally. If the clock is not reset, the device becomes de-registered and a new id token must be obtained to re-register the device. The registration validity period is defined by the start and end date in the ID certificate. Use the **show license tech support** command to get the ID certificate information.

Recommended Action:

Set the system clock back to the correct date and time.

%SMART_LIC-3-ROOT_CERT_MISMATCH_PROD

Message:

```
Certificate type mismatch
```

Explanation:

Smart Agent received an incorrect certificate for validation. Please contact your product support team.

%SMART_LIC-3-APPHA_MISSING_PEER

Message:

```
The Application HA Cluster do not have a member with given identity. Use the 'show license
usage' command to see the exact error.
```

Explanation:

When removing peer information for an entitlement that supports the attribute, the given peer information does not exist. This means that one of the devices may not be configured correctly or that the logic that is supposed to add/update peer information is not working correctly.

%SMART_LIC-3-APPHA_ADD_ITSELF

Message:

```
The Application, is trying to add itself as its own Application HA peer.
```

Explanation:

When adding peer information for an entitlement that supports an attribute, the peer information contains the same data as its own HA attribute. This means that it has tried to add itself as its own peer.

%SMART_LIC-3-CERTIFICATE_VALIDATION

Message:

```
Certificate validation failed by smart agent: \%s
```

Explanation:

The ID certificate validation failed during a reboot, registration or renewal. The included error message should give more information about the failure.

%SMART_LIC-3-HOT_STANDBY_OUT_OF_SYNC

Message:

```
Smart Licensing agent on hot standby is out of sync with active Smart Licensing agent
```

Explanation:

The Smart Licensing Agent on hot standby failed to process the data necessary to stay in sync with the active agent. If a switch over occurs the the new active agent will not be in the same state as the current active agent. The configuration does not match the value of the enable flag in Trusted Store. This can happen if a configuration is copied onto the system and a reload occurs. If the new configuration does not contain the Smart Licensing Enable command, the value in Trusted Store will not match.

%SMART_LIC-3-ENTITLEMENT_RENEW_FAILED

Message:

```
Entitlement authorization with Cisco licensing cloud failed: \%s
```

Explanation:

The device has failed to communicate with Cisco to renew the entitlement authorization.

Recommended Action:

Please verify your Call Home setting and that the device has connectivity to the Cisco Smart Software Manager or satellite

%SMART_LIC-3-COMM_FAILED

Message:

```
Communications failure with the Cisco Smart Software Manager or satellite : \%s
```

Explanation:

The device communication with the Cisco Smart Software Manager or satellite failed.

Recommended Action:

Please verify your Call Home setting and that the device has connectivity to the Cisco Smart Software Manager or satellite

%SMART_LIC-3-CONVERT_FAILED

Message:

```
\%s License conversion failed: \%s
```

%SMART_LIC-3-ID_CERT_RENEW_NOT_STARTED

Message:

```
ID certificate start date not reached yet
```

Explanation:

The device registration failed. The ID Certificate start date is later than the device current time.

Recommended Action:

Please adjust your device clock to be correct, and retry the registration again.

%SMART_LIC-3-ID_CERT_RENEW_FAILED

Message:

```
Automatic registration renewal failed: \%s
```

Explanation:

The automatic ID certificate renewal failed. The included error message should give a better idea of what the failure was.

Recommended Action:

Please verify your Call Home setting and that the device has connectivity to the Cisco Smart Software Manager or satellite

%SMART_LIC-3-EVAL_EXPIRED_WARNING

Message:

```
Evaluation period expired on \%s
```

Explanation:

The device evaluation period will expire in the specified amount of time.

Recommended Action:

Register this device with the Cisco Smart Software Manager or satellite before the evaluation period expires.

%SMART_LIC-3-ROOT_CERT_MISMATCH_DEV

Message:

```
Certificate Mismatch: Development \%s Certificate being used with a Production Root
Certificate. Use the 'test license smart dev-cert enable' CLI to set the DEV root cert.
```

Explanation:

The Production Root Certificate is being used with Development certificates.

Recommended Action:

Please activate the Development Root Certificate from the CLI. (ie. 'test license smart dev-cert enable')

%SMART_LIC-4-CONFIG_NOT_SAVED

Message:

```
Smart Licensing configuration has not been saved
```

Explanation:

This is an informational message to remind you to save the configuration.

Recommended Action:

Save the configuration.

%SMART_LIC-4-HANDLE_ATTR_VERSION_MISMATCH

Message:

```
The handle attribute version between two devices are different. \%s
```

Explanation:

The devices inside a cluster do not have the same operational capability. This is not an issue if all devices only use the functionality that all members of a cluster support. However, it is good practice to have all devices in a cluster using the same software version.

%SMART_LIC-4-RESERVE_IN_PROGRESS

Message: `License Reservation process must be completed with the 'license smart reservation install' command. Reservation started on %s`

Recommended Action:

You must obtain a reservation authorization code from Cisco Smart Software Manager and install it on the device.

%SMART_LIC-4-IN_OVERAGE

Message: `One or more entitlements are in overage`

Explanation:

This is for information only. No action is necessary. You are still in compliance and within the overage amount as specified in your contract.

Recommended Action:

This message is informational only and no action is required.

%SMART_LIC-4-SMART_TRANSPORT_NOT_CONFIG

Message: `Smart Agent for Licensing Smart transport is not configured for utility reporting`

Explanation:

Smart Agent for Licensing Utility is enabled and there is a subscription, but Smart transport is not configured.

%SMART_LIC-4-UTILITY_FQDN_MISMATCH

Message:

```
Smart Agent for Licensing Utility URL setting does not match the FQDN in the utility
certificate.
```

Explanation:

The Smart Agent for Licensing Smart licensing URL must match the FQDN embedded in the utility certificate.

Recommended Action:

Obtain a new utility certificate from Cisco.

%SMART_LIC-4-EVAL_WILL_EXPIRE_WARNING

Message:

```
Evaluation period will expire in \%s.
```

Explanation:

The device is operating within the evaluation period and this period ends in the specified amount of time.

Recommended Action:

Register this device with the Cisco Smart Software Manager or satellite before the evaluation period ends.

%SMART_LIC-4-EVAL_WILL_EXPIRE_WARNING

Message:

```
Evaluation period will expire in \%s.
```

Explanation:

The device is using the evaluation period which will expire in the specified time

Recommended Action:

Register this device with the Cisco Smart Software Manager or satellite before the evaluation period expires.

%SMART_LIC-5-IN_COMPLIANCE

Message: `All entitlements and licenses in use on this device are authorized.`

Explanation:

All your requested entitlements are authorized by Cisco licensing services.

Recommended Action:

This message is informational only and no action is required.

%SMART_LIC-5-COMM_RESTORED

Message:

`Communications with the Cisco Smart Software Manager or satellite restored`

Explanation:

Smart Agent communication with the Cisco Smart Software Manager or satellite has been restored.

Recommended Action:

This is informational only and no action is required

%SMART_LIC-5-SYSTEM_CLOCK_CHANGED

Message:

`Smart Agent for Licensing System clock has been changed`

Explanation:

The system clock has changed and the Smart Agent for Licensing has updated its internal timers

Recommended Action:

This is informational only and no action is required

%SMART_LIC-5-UTILITY_RENEW_SUCCESS

Message:

`Smart Agent for Licensing Utility certificate renewal successful`

%SMART_LIC-5-IN_COMPLIANCE

Message:

`All entitlements and licenses in use on this device are authorized`

Explanation:

All customer requested entitlements are authorized by Cisco licensing services.

Recommended Action:

This is informational only and no action is required

%SMART_LIC-5-EVAL_START

Message:

```
Entering evaluation period
```

Explanation:

The device is not registered with the Cisco Smart Software Manager or satellite and is using licenses. An evaluation period of 90 days is available

Recommended Action:

Register this device with the Cisco Smart Software Manager or satellite using an ID token

%SMART_LIC-5-COMM_INIT_FAILED

Message:

```
Failed to initialize communications with the Cisco Smart Software Manager or satellite: \%s
```

Explanation:

Smart Agent could not initialize communication with the Cisco Smart Software Manager or satellite.

Recommended Action:

Please verify your Call Home setting and check that the device has connectivity to the Cisco Smart Software Manager or satellite.

%SMART_LIC-5-AUTHORIZATION_EXPIRED

Message:

```
Authorization period expired
```

Explanation:

The device has not communicated with the Cisco Smart Software Manager or satellite for 90 days and the device has not automatically renewed the entitlement authorizations. Some features may restrict functionality

Recommended Action:

Please verify your Call Home setting and that the device has connectivity to the Cisco Smart Software Manager or satellite

%SMART_LIC-6-ID_CERT_RENEW_SUCCESS

Message:

```
Automatic registration renewal successful
```

Explanation:

Customer ID certificate has been renewed successfully

Recommended Action:

This is informational only and no action is required

%SMART_LIC-6-DISABLED

Message:

```
Smart Agent for Licensing disabled
```

Explanation:

Smart Agent has been disabled from either the CLI orbecause of a configuration mismatch

%SMART_LIC-6-AUTH_RENEW_SUCCESS

Message:

```
Authorization renewal with the Cisco Smart Software Manager or satellite. State=\%s
```

Explanation:

The automatic authorization renewal was successful

Recommended Action:

This is informational only and no action is required

%SMART_LIC-6-HA_ROLE_CHANGED

Message:

```
Smart Agent HA role changed to \%s.
```

Explanation:

Smart Agent role on HA RP has been changed to either active or standby.

Recommended Action:

This is informational only and no action is required

%SMART_LIC-6-HA_CHASSIS_ROLE_CHANGED

Message:

```
Smart Agent HA chassis role changed to \%s.
```

Explanation:

Smart Agent chassis role on HA has been changed to either active or standby.

Recommended Action:

This is informational only and no action is required

%SMART_LIC-6-AGENT_ALREADY_REGISTER

Message:

```
This device is already registered with the Cisco Smart Software Manager or satellite.
```

Explanation:

Smart Licensing on this device has already registered with the Cisco Smart Software Manager or satellite

Recommended Action:

Use the force option when registering or remove this device from your virtual account on the Cisco Smart Software Manager or satellite

%SMART_LIC-6-AGENT_ALREADY_DEREGISTER

Message:

```
Smart Agent is already Deregistered with the CSSM.
```

Explanation:

Smart Licensing has already de-registered with Cisco.

%SMART_LIC-6-EXPORT_CONTROLLED

Message:

```
Usage of export controlled features is \%s
```

Explanation:

This tells you if you are allowed to use export controlled features.

Recommended Action:

This is informational only and no action is required.

%SMART_LIC-6-HOSTNAME_MATCHED_UDI

Message:

```
The host name has been changed to match a field in the device identifier (UDI).Since the
device identifier is sent to Cisco this may bypass your host name privacy settings
```

Explanation:

The host name has been changed to match a field in the device identifier (UDI). Since the device identifier is sent to Cisco this may bypass your host name privacy settings.You can view the device identifier using the command: **show license udi**.

Recommended Action:

Change the host name so it does not include any fields in the device identifier.

%SMART_LIC-6-RESERVED_INSTALLED

Message:

```
\%s License Reservation Authorization code installed
```

Recommended Action:

This is informational only and no action is required.

%SMART_LIC-6-ENTITLEMENT_RENEW_SUCCESS

Message:

```
Entitlement authorization renewal with Cisco licensing cloud successful
```

Explanation:

Authorization renewal request is successful.

Recommended Action:

This is informational only and no action is required

%SMART_LIC-6-RESERVE_RETURNED

Message:

```
\%s License Reservation returned. Smart Agent is now unregistered.
```

Recommended Action:

This is informational only and no action is required.

%SMART_LIC-6-RESERVE_CANCELED

Message:

```
\%s License Reservation request canceled. Smart Agent is now unregistered.
```

Explanation:

Sent when you cancel a reservation request by using the **reservation cancel** command.

Recommended Action:

This is informational only and no action is required

%SMART_LIC-6-RESERVE_AUTH_FAILED

Message:

```
Failed to validate the \%s Reservation Authorization Code. Changing to the unregistered
state.
```

Explanation:

The reservation authorization code is not valid on this device

%SMART_LIC-6-RESERVE_HA_MISMATCH

Message:

The reserved licenses on the active and standby do not match. Use the `show license status` command to see the error details.

Explanation:

The Licenses reserved using the Specified License Reservation (SLR) feature in Smart Licensing and installed on the active and standby or member devices in an HA configuration are not the same. If the standby takes over as active you will not have the same licenses available and your device may not work properly.

%SMART_LIC-6-PLR_DISABLED_INIT_COMM

Message:

```
Permanent License Reservation has been disabled. Please reboot the system to initialize
Smart Licensing communications with Cisco.
```

Explanation:

During bootup, Smart Licensing communication is not initialized if Permanent License Reservation (PLR) is enabled. To enable Smart Licensing communication with Cisco when PLR is disabled, the system needs to be rebooted.

%SMART_LIC-6-CONVERT_START

Message:

```
Smart License Conversion has started
```

%SMART_LIC-6-CONVERT_LIC_SUCCESS

Message:

```
\%s License \%s has been converted to \%s with a count of \%d
```

%SMART_LIC-6-CONVERT_LIC_ALREADY

Message:

```
\%s License \%s has been converted to \%s with a count of \%d
```

%SMART_LIC-6-CONVERT_SUCCESS

Message:

```
\%s Smart License Conversion successful
```

%SMART_LIC-6-CONVERT_ALREADY

Message:

```
\%s Smart License Conversion successful
```

%SMART_LIC-6-THIRDPARTY_MODE_ENABLED

Message:

```
Smart Agent for Licensing is in Thirdparty Mode
```

Explanation:

Smart Agent for Licensing is in thirdparty mode, and ready to collect and process RUM reports

%SMART_LIC-6-THIRDPARTY_MODE_DISABLED

Message:

```
Smart Agent for Licensing is out of Thirdparty Mode
```

Explanation:

Smart Agent for Licensing is out of thirdparty mode, and has stopped collecting and processing RUM reports.

%SMART_LIC-6-UTILITY_STARTED

Message:

```
Smart Agent for Licensing Utility has started sending usage reports
```

Explanation:

Smart Agent for Licensing utility has been enabled and is sending usage reports.

%SMART_LIC-6-UTILITY_STOPPED

Message:

```
Smart Agent for Licensing Utility has stopped sending usage reports: \%s
```

Explanation:

Smart Agent for Licensing Utility is not available and no longer sending usage reports.

%SMART_LIC-6-AGENT_READY

Message:

```
Smart Agent for Licensing is initialized
```

Explanation:

Smart Agent for Licensing is fully initialized and ready for use.

Recommended Action:

This is informational only and no action is required

%SMART_LIC-6-AGENT_ENABLED

Message:

```
Smart Agent for Licensing is enabled
```

Explanation:

Smart Agent for Licensing is enabled and ready to process licensing requests.

Recommended Action:

This is informational only and no action is required

%SMART_LIC-6-AGENT_REG_SUCCESS

Message:

```
Smart Agent for Licensing Registration with the Cisco Smart Software Manager or satellite
```

Explanation:

Smart Licensing registration was successful.

%SMART_LIC-6-AGENT_DEREG_SUCCESS

Message:

```
Smart Agent for Licensing De-registration with the Cisco Smart Software Manager or satellite
was successful
```

Explanation:

Smart Licensing de-registration successful.

Recommended Action:

This is informational only and no action is required

%SMART_LIC-7-DAILY_JOB_TIMER_RESET

Message:

```
Daily job timer reset
```

Explanation:

This message is used only for testing and does not indicate an error

Recommended Action:

This is informational only and no action is required

Registering the Router with the Cisco Licensing Cloud

Note If you are registering the router and using CSSM satellite, go to the following section instead: Registering the Router with the Cisco Licensing Cloud (CSSM satellite), on page 202.

After you have enabled Cisco Smart Licensing, you must register the router with Cisco. Using the ID token, the license agent on the router registers the product with Cisco and then receives back an identity certificate. This certificate is used for all future communications with Cisco. The license agent on the router automatically renews the registration information with Cisco every 30 days. This registration step is performed once for each product instance.

Procedure

	Command or Action	Purpose

Step 1	**license smart register idtoken** *id-token*
	Example:
	`Router# license smart register idtoken YjBkOWM5YTItMDFiOS00ZjBmLTllY2YtODEzMzg1`
	`YlMyZDVhLTExODEtMjE0%0ANzc5NDF8UlBUTAySWERTmJqalNnbmlzRUIyaGlYU053L0pHZlNvUW9V`
	`TFpE%0AekxCOD0%3D%0A`
	`The system will now contact the Cisco Smart Licensing servers`
	`to obtain authorization for Smart Licensing`
		Registers the device instance with the Cisco licensing cloud. This step only needs to be performed once per device instance.
		The license agent registers the product with Cisco and receives back an identity certificate. This certificate is saved and automatically used for all future communications with Cisco. The license agent automatically renews the registration information with Cisco every 30 days.

The device is registered with Cisco.

Note Smart licensing may fail when IPv6 is configured on any interface and the device does not have IPv6 connectivity to the Internet or the Cisco Smart Software Agent at tools.cisco.com, resulting in log file error messages such as those shown below.

(Note that these messages may also appear as a result of other conditions being true.)

```
%SMART_LIC-3-AGENT_REG_FAILED: Smart Agent for Licensing Registration with Cisco licensing
 cloud failed: Fail to send out Call Home HTTP message.

%SMART_LIC-3-COMM_FAILED: Communications failure with Cisco licensing cloud: Fail to send
out Call Home HTTP message.
```

If connectivity failed due to this issue, see Re-establishing Connectivity to the Cisco Smart Call Home Server when IPv6 is Configured, on page 204.

What to do next

Go to Requesting Cisco Smart License Throughput Level Licenses, on page 204.

Registering the Router with the Cisco Licensing Cloud (CSSM satellite)

Note If you are registering the router and using CSSM satellite, go to the following section instead: Registering the Router with the Cisco Licensing Cloud, on page 201.

After you have enabled Cisco Smart Licensing, you must register the router with Cisco. Using the ID token, the license agent on the router registers the product with Cisco and then receives back an identity certificate. This certificate is used for all future communications with Cisco. The license agent on the router automatically renews the registration information with Cisco every 30 days. This renewal of registration is done once for each product instance. See http://www.software.cisco.com to determine the id-token.

SUMMARY STEPS

1. **profile** CiscoTAC-1
2. **no destination address http** *default-url*
3. **destination address http** *satellite-url*
4. **exit**
5. **crypto pki trustpoint** SLA-TrustPoint
6. **revocation-check none**
7. **exit**
8. **license smart register idtoken** *id-token*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**profile** CiscoTAC-1 **Example:** `Router(cfg-call-home)# profile CiscoTAC-1`	Enables TAC profile configuration mode.
Step 2	**no destination address http** *default-url* **Example:** `Router(cfg-call-home-profile)# no destination address https://tools.cisco.com/its/service/oddce/services/DDCEService`	Removes the previously configured destination address for the Cisco Smart Software Agent.
Step 3	**destination address http** *satellite-url*	*satellite-url*—To determine the URL of the transport gateway, see the CSSM Satellite documentation. The *satellite-url* is similar to this example: `http://<ip-address>/Transportgateway/services/DeviceRequestHandler`
Step 4	**exit**	Exits TAC profile configuration mode.
Step 5	**crypto pki trustpoint** SLA-TrustPoint	Starts ca-trustpoint configuration mode and create a name, SLA-Trustpoint, for the CertificateAuthority server.
Step 6	**revocation-check none**	Certificate checking is ignored. Use this command if you are configuring software using Cisco Smart Software Manager satellite (CSSM satellite). This command ensures that revocation checking of the certificate is disabled when the trust policy is in use.
Step 7	**exit**	Exits ca-trustpoint configuration mode and returns to global configuration mode.

Step 8	**license smart register idtoken** *id-token*
	Example:
	`Router# license smart register idtoken YjBkOWM5YTItMDFiOS00Z`
	`jBmLTllY2YtODEzMzg1YTMyZDVhLTEzODE0MjE0%0ANzc5NDF8U1B`
	`DUTAySWFRImJqalNnbmlzRUIyaGlYU053L0pHZTNvUW9VTFpE%0AekxCOD0%3D%0A`
	The system now contacts the Cisco Smart Licensing servers to obtain authorization for Smart Licensing
		Registers the device instance with the Cisco licensing cloud. This step only needs to be performed once per device instance.
		The license agent registers the product with Cisco and receives back an identity certificate. This certificate is saved and automatically used for all future communications with Cisco. The license agent automatically renews the registration information with Cisco every 30 days.

What to do next

Go to Requesting Cisco Smart License Throughput Level Licenses, on page 204

Re-establishing Connectivity to the Cisco Smart Call Home Server when IPv6 is Configured

This section describes what to do if there is a failure to connect to the Cisco Smart Call Home Server when IPv6 is configured, as mentioned previously in Enabling Cisco Smart Licensing, on page 181.

To re-establish connectivity with the Cisco Smart Call Home Server, use one of the following two methods, depending on the version of Cisco IOS XE that you are using:

If you are using one of the following recent versions of Cisco IOS XE: 3.16.6, Denali 16.3.4 and later, Everest 16.4.2 and later, Everest 16.5.1 and later, see Re-establishing Connectivity, on page 204.

Note that in some cases, after configuring the previous steps, you may need to restart the router to fully re-establish connectivity.

Re-establishing Connectivity

This method applies to the following Cisco IOS XE releases: Cisco IOS XE 3.16.6, Cisco IOS XE Denali 16.3.4 and later, Everest 16.4.2 and later, and Everest 16.5.1 and later.

If there is an IPv6 address on an interface and the device cannot connect to the Internet or Smart software agent, configure the interface to only use IPv4 for smart licensing, with the following configuration mode command:

Procedure

ip http client source-interface *interface*

Example:

Router(config)# **ip http client source-interface** GigabitEthernet1

Note
The interface GigabitEthernet1 needs to have an IPv4 address, not an IPv6 address.

Configures the interface to use IPv4.

Note
The call-home profile configuration with the static IP address corresponding to FQDN *tools.cisco.com* is not recommended as a long-term solution, since the IP address might change in future.

Requesting Cisco Smart License Throughput Level Licenses

Request a license corresponding to the configured technology package level and throughput level.

Prerequisites

Register the device with the Smart License server.

Changing Throughput Licenses

When working with Cisco Smart Licenses, using the **platform hardware throughput level** command requests a license for the new throughput level. Typically, the activation process requires several minutes. During this time, the new license remains "pending."

Before the Cisco IOS XE 3.17 release, when changing throughput, the effective throughput would drop to 100 kbps while the new throughput license was pending. When the new license was activated, throughput would change to the newly configured level.

Beginning in the Cisco IOS XE 3.17 release, and including Cisco IOS XE Denali 16.2 and later, the transition is smoother. The router maintains the original throughput level until the license for the new throughput is activated.

In the following example, the router has been authorized previously for a throughput of 100M. The first line in the example is a request for a 250M throughput license. While the request is pending, the **show license all** command indicates the current authorized level (100M) and the pending license (250M), both shown in bold.

```
ultra-mcp(config)#platform hardware throughput level MB 250
Wait for 250M license request to succeed, continue to use existing 100M license until then
ultra-mcp(config)#end
ultra-mcp#show license all
Smart Licensing Status
======================
Smart Licensing is ENABLED
Registration:
  Status: REGISTERED
  Smart Account: CSR1000v
  Virtual Account: AX_SEC_IPB
  Export-Controlled Functionality: Allowed
  Initial Registration: SUCCEEDED on Nov 06 11:59:12 2015 UTC
  Last Renewal Attempt: None
  Next Renewal Attempt: May 04 11:59:11 2016 UTC
  Registration Expires: Nov 05 11:56:09 2016 UTC
License Authorization:
  Status: AUTHORIZED on Nov 09 13:37:00 2015 UTC
  Last Communication Attempt: SUCCEEDED on Nov 09 13:37:00 2015 UTC
  Next Communication Attempt: Nov 09 13:39:20 2015 UTC
  Communication Deadline: Feb 07 13:33:58 2016 UTC
License Usage
==============
regid.2014-05.com.cisco.ax_100M
,1.0_2fff5ed6-e23c-455d-ade3-83ba3c8ed890 (ax_100M):
  Description:
  Count: 1
  Version: 1.0
  Status: AUTHORIZED
(ax_250M
):
  Description:
  Count: 1
  Version: 1.0
  Status: PENDING

Product Information
===================
UDI: PID:CSR1000v,SN:9R8ORIT8CB0
Agent Version
=============
Smart Agent for Licensing: 1.4.0_rel/28
Component Versions: SA:(1_4_rel)1.1.7, SI:(rel22)1.1.0, CH:(rel5)1.0.1, PK:(rel18)1.0.0
```

SUMMARY STEPS

1. **configure terminal**
2. **license boot level** {**ipbase** | **security** | **ax** | **appx**}
3. **platform hardware throughput level MB** {**10** | **100** | **1000** | **10000** | **250** | **2500** | **50** | **500** | **5000** }

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal**	Enter config mode.
Step 2	**license boot level** {**ipbase** | **security** | **ax** | **appx**}	Specify the technology package level.
Step 3	**platform hardware throughput level MB** {**10** | **100** | **1000** | **10000** | **250** | **2500** | **50** | **500** | **5000** }	Configure the throughput level for the license to request.

Requesting Memory Add-on License

For information about memory add-on licenses, see Understanding the Cisco CSR 1000v Memory Allocation, on page 168. For Cisco Smart Licensing, the procedure for requesting the license is as follows:

SUMMARY STEPS

1. **configure terminal**
2. **platform memory add** *memory*
3. **show platform software vmemory info**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal**	Enter config mode.
Step 2	**platform memory add** *memory*	
Step 3	**show platform software vmemory info**	Verifies the updated memory allocation.

Requesting Smart License Broadband license

For information about broadband licenses, see Information About Installing Broadband Feature License, on page 173 and Installing Broadband Feature License, on page 174. For Cisco Smart Licensing, the procedure for requesting the license is as follows:

SUMMARY STEPS

1. **configure terminal**
2. **platform broadband** {**1K** | **2K** | **3K** | **4K**}

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal**	Enter config mode.
Step 2	**platform broadband** {**1K** | **2K** | **3K** | **4K**} **Example:** `Router(config)# platform broadband 1K`	Adds support for the number of broadband sessions to accommodate the added broadband feature license(s). You can add 1000 sessions for each broadband feature license you are planning to install. For example, if you plan to add two broadband feature licenses, enter the value as 2K.

Manually Renewing the ID Certificate

By default, the ID certificate is automatically renewed every 6 months. You can manually renew the ID certificate using this procedure.

This may be useful in either of the following circumstances:

- If you have a limited window of Internet access
- After making licensing changes in the Smart Software Manager

SUMMARY STEPS

1. **license smart renew id**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**license smart renew id** **Example:** `Router# license smart renew id`	Renews the ID certificate.

Manually Renewing the License

By default, the license (also called “entitlement”) is automatically renewed every 30 days. You can manually renew the license using this procedure.

This may be useful in either of the following circumstances:

- Only a limited window of Internet access is available.
- After making licensing changes in the Smart Software Manager.

Note The terms “license” and “entitlement” are equivalent and are used interchangeably.

SUMMARY STEPS

1. **license smart renew auth**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**license smart renew auth** **Example:** `Router# license smart renew auth`	Renews the license (also called “entitlement”).

Unregistering a Device from Cisco Smart Licensing

SUMMARY STEPS

1. **license smart deregister**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**license smart deregister**	Removes the Cisco Smart Licensing registration for the device instance. All Cisco Smart Licensing certificates are removed on the router and the entitlements are released from the Smart Call Home backend server.

Disabling Cisco Smart Licensing

Describes how you can disable Cisco Smart Licensing and switch back to standard Cisco Software Licensing (CSL) mode.

SUMMARY STEPS

1. **no license smart enable**
2. **reload**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**no license smart enable**	Disables Cisco Smart Licensing on the device instance and switches to Cisco Software Licensing (CSL) mode. Reboot the device for the change to take effect. **Note**: When you disable Smart Licensing, the Cisco Software License (CSL) and all licensing calls pass through the Smart Agent. For the **no** case, if Smart Licensing is already registered, the Smart Agent performs the **license smart deregister** operation that deactivates Smart Licensing.
Step 2	**reload**	Restarts the router. This is required to complete the process of disabling the Cisco Smart License.

License Out-of-Compliance Behavior

A successfully licensed router may receive an "out of compliance" syslog message during reload or renewal/reauthorization of license from the Smart Licensing server if an attempt is made to contact the Smart Licensing server for a license and the number of available licenses recorded on the Smart Licensing account is exceeded. This message may also occur as a result of the router having been configured to have a higher performance level compared to the previously purchased feature set.

After the "out of compliance" message appears, the system continues to operate at the previously licensed throughput rate.

License Behavior with no Connectivity to the Smart Licensing Server

When a successfully licensed Cisco CSR 1000v/Cisco ISRv is unable to contact the Smart Licensing server, during reload or reauthorization or renewal of license, then the router continues to operate at the previously licensed state.

The license authorization expires if the CSSM satellite server has had no connectivity with the Smart Licensing server for more than 90 days. The license then changes to a **License Authorization Expired State** and the router continues to operate at the previously licensed state and runs in the Feature Restricted mode.

The following example shows a typical license expiry message that appears on the console.

```
*Aug 4 08:02:19.056: %VXE_THROUGHPUT-6-CLI_RESTRICTED_LICENSE_EXPIRE: System is in
feature restricted mode due to license expire. Configuration CLIs have been blocked.
nvram:startup-config is write protected (read-only). Valid license and reboot is required
to recover
from this state. Use configuration CLI - platform hardware throughput
restricted-throughput-rate-mode
if startup-config changes are needed.
```

In the Feature Restricted mode, the feature configuration commands are blocked except for those commands that are needed for licensing. Also, the commands for setting the technology features and the throughput rate are available.

Note In the Feature Restricted mode you cannot save or write the running configuration. However, starting from the 17.1.1 release, you can execute the copy command except while copying to statup config/NVRAM, as startup configuration is write protected.

From the 17.1.1 release, when your device is in the Feature Restricted mode, if the hostname was previously Router#, it changes to (restricted)Router#. Renew your license to move out of the Feature Restricted mode.

Example 1

This example shows that if you enter a write command an error message appears.

router# **write**

```
nvram config write protected
```

Example 2.

This example shows that if you enter a reload command an error message appears.

router# **reload**

```
System configuration has been modified. Save? [yes/no]: yes
nvram config write protected
Proceed with reload? [confirm]
```

After you confirm by pressing Enter and proceed with the reload, the existing configuration is retained and the router continues to run. Note that the existing configuration cannot be modified.

If you further attempt to authorize or renew licenses with the Smart Licensing server, and if communication cannot be achieved with the Smart Licensing server (e.g, after rebooting the router), the syslog messages about being in the Feature Restricted mode, are generated. The following example displays this scenario:

```
*Aug 4 08:02:19.056: %VXE_THROUGHPUT-6-CLI_RESTRICTED_LICENSE_EXPIRE:

System is in feature restricted mode due to communication fault to license server.
Configuration CLIs have been blocked. nvram:startup-config is write protected (read-only).
Valid license and reboot is required to recover from this state.
Use configuration CLI - platform hardware throughput restricted-throughput-rate-mode
if startup-config changes are needed.
```

In the Feature Restricted mode, the **platform hardware throughput restricted-throughput-rate-mode** command is enabled. This command is only visible in the Feature Restricted mode. After issuing this command, the throughput rate becomes 100 Kbps after the next reload. See the following example:

```
router(config)# platform hardware throughput restricted-throughput-rate-mode
```

After you enter the **platform hardware throughput restricted-throughput-rate-mode** command, the system displays the following message:

```
% The config will take effect on next reboot. This device will need to be re-licensed
```

After the next reload, the throughput rate restricted mode is in operation, and the traffic throughput rate is 100 Kbps.

When the modification of the start-up configuration is disabled, changes you make (such as registering a new license or configuring the license server connectivity) in the Feature Restricted mode are lost if you reboot the router. The router then returns to the Feature Restricted mode. If you use the **platform hardware throughput restricted-throughput-rate-mode** command, recovery is possible as you can use the configuration commands that are required to restore the router license.

If the configuration commands to renew licensing are not required, you can reboot the Cisco CSR 1000v / Cisco ISRv instance. The device communicates with the Smart Licensing server and then pre-existing licenses are renewed.

Activating Permanent License Reservation

Introduction to Activating Permanent License Reservation

Activating a license using Permanent License Reservation (PLR) allows a device to use a license without having to be connected to Cisco Smart Software Manager (CSSM) or CSSM satellite. This feature is available using release Cisco IOS XE Everest 16.5.1a or later.

Activating Permanent License Reservation

This process describes how to activate permanent license reservation (PLR), which allows a product instance or device to have universal entitlement to a license. After activating PLR, the device does not need to communicate with Cisco Smart Software Manager(SSM) or a Cisco SSM server to maintain its ability to use features associated with a license. The license will not become out of compliance.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal**	Enter global configuration mode on the device onto which you want to install a license using PLR.
Step 2	**license smart enable**	Enable smart licensing.
Step 3	**license smart reservation**	Enables reservation mode.
Step 4	**end**	

Step 5	**license smart reservation request universal**	Request a reservation request code, which you will later enter in Cisco SSM. For example:
		`license smart reservation request universal`
		`Enter this request code in the Cisco Smart Software Manager portal:`
		`BC-ZCSR1000V:959Z2A5VVWQ-AB5nwN3rt-09`
		Make a note of the request code.

Step 6	Log into Cisco Smart Software Manager (SSM) and navigate to the virtual account that contains the licenses that you need.	**Note** In the list of licenses listed in the virtual account, ensure that there is a permanent license reservation (PLR) license available. (Click the license name and see if the license expiration is "Perpetual".)
Step 7	In Cisco SSM, in the **Licenses** tab, click **License Reservation** and enter the request code obtained in Step 5. The authorization code is displayed on the screen (short ASCII character string).	
Step 8	Make a note of the authorization code.	The authorization code is securely tied to the Universal Device Identifier (UDI) of the device.
Step 9	Enter global config (EXEC) mode on the device where you want to activate PLR.	
Step 10	**license smart reservation install** *auth-code*	(*auth-code* is the authorization code obtained in Step 7.) PLR is now active. This reservation is permanent until it is manually returned or deactivated after it is no longer required.
Step 11	**platform hardware throughput level MB** *throughput*	Set the throughput level (MB) (Range 10 to 10000).

What to do next

Later, to return or deactivate PLR, see Deactivating Permanent License Reservation, on page 213.

Deactivating Permanent License Reservation

Follow these steps to deactivate (return) Permanent License Reservation (PLR). For example, if you want to change the licensing on the device and save the cost of using PLR.

Procedure

Step 1 Enter EXEC mode.

Step 2 **license smart reservation return** {*auth-code*}

- **license smart reservation return**
- **license smart reservation return auth-code**

Note
In rare cases, you can use this second form of the command, when you input a value for the *auth-code*. This is useful when the current reservation status of the license was previously cleared; for example, after the **license smart reservation cancel** command was used to cancel a request.

This command generates a reservation return code.

Example:

```
license smart reservation return
Reservation return code: BAAeUF-rz6EiG-PXLMQB-CRBnrx-TsaEep-A3x
```

Step 3 Make a note of the return code.

Step 4 In CSSM, log in and navigate to the **Product Instances** tab for the virtual account in which the product is registered. Locate the entry that matches the Unique Device Identifier (UDI) of the device that you want to remove from the list.

Step 5 Select **Actions**, click **Remove Product Instance**.

Cisco SSM removes the product instance.

Step 6 Paste the return code and click **Remove**.

The device is unregistered and operates with a default throughput of 1 Mbps.

Enabling Utility Reporting

Utility Reporting—Overview

Utility Reporting allows you to pay for features based on usage, instead of paying in advance for feature licenses. Utility Reporting is available for the Cisco CSR 1000v or Cisco ISRv using Cisco IOS XE Fuji 16.8.1 or higher.

Utility Reporting collects usage data from products that have Cisco Smart Licensing and Utility Reporting enabled and sends the usage data via the CSSM satellite to the Cisco Service Billing Platform (SBP), which

produces daily reports. Usage data is produced in the Resource Utilization Measurement (RUM) format (ISO/IEC 19770-4).

The device collects usage data every 15 minutes and sends it to the CSSM satellite every four hours. Every 8 hours the CSSM satellite sends data to the Cisco SBP. The device stores usage data for up to 30 days and the CSSM satellite saves data for 90 days. This data backup allows recovery after, for example, the CSSM satellite becomes disconnected from the Cisco SBP. If there is a lack of connection for more than 30 days, the Cisco accounts team should contact you to communicate any issues.

Utility Reporting—Prerequisites

Before enabling Utility Reporting, perform the following steps:

- Add the licensese that are going to be used for utility reporting (using a post-payment method) into a smart account. For example, see Smart Software Licensing Overview.
- Enable Smart Licensing on the device.

  Summary steps:

  ```
  configure terminal
  license smart enable
  exit
  ```

  See Enabling Cisco Smart Licensing, in this document.
- Install and configure CSSM satellite. For further information, see Smart Software Manager satellite.
- Register the device. For further information, see Registering the Router with the Cisco Licensing Cloud (CSSM satellite), on page 202, in this document.

How to Enable Utility Reporting

Procedure

Step 1 **configure terminal**

Enters the global configuration mode.

Step 2 **license smart utility**

Registers your intention to use the Utility Reporting feature on the device.

Use the "no" form of this command, `no license smart utility`, to signal your intention to remove the Utility Reporting feature.

Example:

```
Device(config)# license smart utility
```

Step 3 **license smart transport smart**

Sets the transport type as `smart` (4th keyword above). This is required for utility reporting communication between the smart agent and the CSSM or the CSSM satellite.

Example:

```
Device(config)# license smart transport smart
```

Step 4 **license smart url** *registration-url*

Sets the URL to be used by the smart transport between the smart agent and the CSSM or the CSSM satellite. Here,

registration-url - is the transport gateway URL. For more information, see CSSM Satellite. *registration-url* has the following form: `https://<OnPrem-FQDN>/SmartTransport`

Use the "no" form of this command, `no license smart url`, to clear the value of the *registration-url*.

Example:

```
Device(config)# license smart url https://cssm-onprem.dcloud.cisco.com/SmartTransport
```

Verifying Utility Reporting

To verify that utility reporting data is being sent from the CSSM satellite to Cisco, enter the command shown in the following example, in configuration mode:

show license all

In the sample output shown below, in the "`Utility`" section, the line "`Last attempt: SUCCEEDED on Dec 19 18:23:02 2017 UTC`" shows when the utility report data was last sent successfully. Other data shows the date and time that the report was sent, and the expected date and time of the next utility report (4 hours later).

```
Smart Licensing Status
======================

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Export-Controlled Functionality: Allowed
  Initial Registration: SUCCEEDED on Dec 13 16:11:51 2017 UTC
  Last Renewal Attempt: None
  Next Renewal Attempt: Feb 19 16:24:44 2018 UTC
  Registration Expires: Jul 05 16:50:33 2018 UTC

License Authorization:
  Status: AUTHORIZED on Dec 13 18:23:51 2017 UTC
  Last Communication Attempt: NOT STARTED
    Failure reason: Device in Thirdparty Utility Mode
  Next Communication Attempt: None
  Communication Deadline: Mar 19 18:23:02 2018 UTC

Utility:
  Status: ENABLED
  Utility report:
    Last success: Dec 19 18:23:03 2017 UTC
    Last attempt: SUCCEEDED on Dec 19 18:23:02 2017 UTC
    Next attempt: Dec 19 22:23:02 2017 UTC

  Customer Information:
    Id: <empty>
    Name: <empty>
    Street: <empty>
    City: <empty>
    State: <empty>
    Country: <empty>
```

```
    Postal Code: <empty>

Data Privacy:
  Sending Hostname: yes
    Callhome hostname privacy: DISABLED
    Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED

Transport:
  Type: Smart
  Registration URL: https://cssm-onprem.dcloud.cisco.com/SmartTransport
  Utility URL: http://10.87.9.106/Transportgateway/services/DeviceRequestHandler

License Usage
==============

(ax_1G):
  Description:
  Count: 1
  Version: 1.0
  Status: AUTHORIZED
  Utility Subscription id:  81

Product Information
===================
UDI: PID:CSR1000V,SN:9ZT6BKI8CXG

Agent Version
=============
Smart Agent for Licensing: 4.3.0_rel/8
Component Versions: SA:(1_3_dev)1.0.15, SI:(dev22)1.2.1, CH:(rel5)1.0.3, PK:(dev18)1.0.3
```

To set the information that will appear in the "`Customer Information`" section of the **show license** command above, use the following command:

license smart utility customer_info *info_type info_value*, where info_type is one of the following: city, country, id, name, postalcode, state, street.

Example

```
license smart utility customer_info city New York
```

Troubleshooting Cisco Smart License Issues

Determining Device Registration Information

Use the **show license all**command to display the device registration information.

```
Router#show license all
Smart Licensing Status
======================
Smart Licensing is ENABLED
Registration:
  Status: REGISTERED
  Smart Account: BU Production Test
  Virtual Account: CRDC_SH_3
  Export-Controlled Functionality: Allowed
  Initial Registration: SUCCEEDED on Jul 08 20:45:54 2015 UTC
  Last Renewal Attempt: None
```

```
  Next Renewal Attempt: Jan 04 20:45:54 2016 UTC
  Registration Expires: Jul 07 05:59:29 2016 UTC
License Authorization:
  Status: AUTHORIZED on Jul 08 20:46:05 2015 UTC
  Last Communication Attempt: SUCCEEDED on Jul 08 20:46:05 2015 UTC
  Next Communication Attempt: Aug 07 20:46:05 2015 UTC
  Communication Deadline: Oct 06 05:59:43 2015 UTC
License Usage
==============
CSR 1KV AX 500M (ax_500M):
  Description: CSR 1KV AX 500M
  Count: 1
  Version: 1.0
  Status: AUTHORIZED
Product Information
===================
UDI: PID:CSR1000v,SN:9Q0BWG3BHL0
Agent Version
=============
Smart Agent for Licensing: 1.4.0_rel/11
Component Versions: SA:(1_4_rel)1.0.10, SI:(rel21)1.2.0, CH:(rel4)1.0.23, PK:(rel17)1.0.5
```

Additional Commands for Troubleshooting

The **show call-home profile all** and **show license tech support** commands may be helpful during troubleshooting.

Understanding the License-Based Restriction on Aggregate Bandwidth

The router includes a license shaper that may restrict the aggregate bandwidth of the router's interfaces. For example, if a 50 Mbps license is installed, then a maximum of 50 Mbps of bidirectional traffic is possible.

The license shaper regulates the throughput on interfaces for both priority traffic and non-priority traffic:

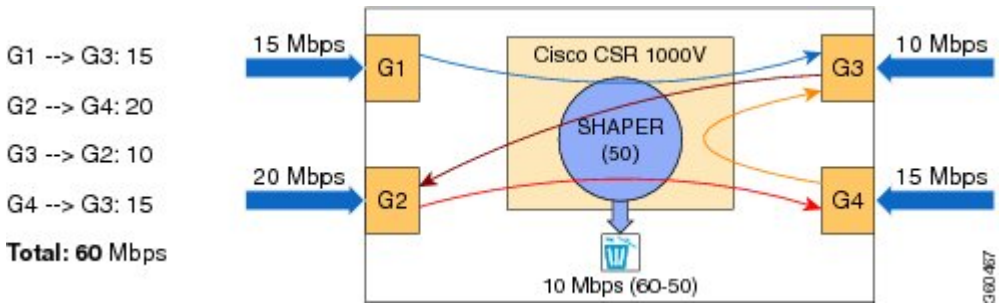
- (Cisco IOS XE 3.10S and earlier) The license shaper regulates the throughput on non-management interfaces only. The GigabitEthernet 0 dedicated management interface is not counted

  .
- (Cisco IOS XE 3.11S and later, and IOS XE Denali 16.2 and later) The license shaper regulates the throughput on all interfaces.

Throughput limits are checked globally, not on a per-interface basis. The license shaper does not distinguish between different types of traffic, such as for IPSec or NAT. If the throughput level is exceeded, then packets may get discarded.

The figure below shows how the license shaper, also known as a traffic shaper, works. In this example, the four interfaces on a Cisco CSR 1000v are passing an aggregated traffic level of 60 Mbps. Because this exceeds the 50 Mbps license-enforced maximum throughput, 10 Mbps of traffic is discarded.

Figure 6: Throughput Example



To check the license-based performance limiter value, use the following command for your interface:

```
Router# show platform hardware qfp active feature qos queue out int GigabitEthernet1 hier
det | inc max:
     orig_max  : 0   ,  max: 33333          child policy-map
     orig_max  : 0   ,  max: 500000         parent policy-map
     orig_max  : 0   ,  max: 1050000000     interface rate limiter
     orig_max  : 0   ,  max: 2500000        license performance limiter
     orig_max  : 0   ,  max: 10000000000    entry for ROOT/SIP infra (ignore rate)
```

The value for the license performance limiter field should match the current maximum throughput level as shown with the **show platform hardware throughput level** command.

Note The license shaper includes an extra scheduler node in the default HQF hierarchy. The router does not provide an option to detect congestion for a particular node in the HQF hierarchy.

For more information about verifying the VM performance indicators, see your hypervisor documentation.

To verify the actual throughput, use the following command:

```
Router# show platform hardware qfp active datapath utilization summary
      CPP 0:                 5 secs    1 min     5 min     60 min
      Input:   Total (pps)    59232     59234     59237     59234
                     (bps)    58757104  58757824  58760840  58757880
      Output:  Total (pps)    48839     48835     48833     48833
                     (bps)    50011264  50012072  50009312  498768736
      Processing: Load (pct)  33        34        34        34
```

In the example, the input rate shown in bold is close to 60 Mbps. The output rate shown in bold is close to 50 Mbps. In this case, the input rate exceeds 50 Mbps, the maximum license rate allowed.

The following command displays the number of packages dropped when the maximum throughput is exceeded:

```
Router# show platform hardware qfp active statistics drop clear | exc _0_
-------------------------------------------------------------------------
Global Drop Stats                Packets           Octets
-------------------------------------------------------------------------
TailDrop                         2018258           256333010
```

When the actual throughput level approaches the maximum allowed by the installed license, you will receive an alert message similar to the following (the message may differ depending on the release version):

```
Dec 13 22:00:29.699: %BW_LICENSE-3-THROUGHPUT_THRESHOLD_LEVEL: F0: cpp_ha:
```

```
Average throughput rate exceeded 95 percent of licensed bandwidth 3 times, sample period
300 seconds, in last 24 hours
```

When the throughput exceeds the maximum allowed bandwidth set by the license, you will receive an alert message similar to the following (Cisco IOS XE 3.12S and later):

```
*Dec 13 22:00:29.699: %BW_LICENSE-4-THROUGHPUT_MAX_LEVEL: F0: cpp_ha:
Average throughput rate exceeded the total licensed bandwidth 50000000 bps and dropped 7
times, sample period 300 seconds, in last 24 hours
```

You can configure the QoS policies at the interface level to guarantee that high-priority traffic is not dropped. For more information, see the QoS configuration guides: https://www.cisco.com/c/en/us/support/routers/cloud-services-router-1000v-series/products-installation-and-configuration-guides-list.html.

Managing Throughput Notifications

You can configure the Cisco CSR 1000v/ ISRv to notify you when a certain percentage of the maximum throughput level is reached. The maximum allowable throughput is based on the installed throughput license.

Note This feature is available on Cisco IOS XE 3.13S or later, and Cisco IOS XE Denali 16.3.1 and later.

By default, when the router first boots, the throughput level notification is enabled, and notifications are sent when the router reaches 95 percent of the maximum throughput level. The throughput level is measured every 300 seconds. When the router is rebooted, the threshold and interval level settings configured using the set platform hardware throughput-monitor command are retained.

The following command configures the hardware throughput monitor settings. The **threshold** *percentage* value represents the percentage of the maximum throughput at which the system notifies you. The valid range is from 75 to 95, and the default value is 95 percent.

The **interval** value represents how often the system measures the throughput level. The valid range is from 30 to 86400 seconds. The default value is 300 seconds.

set platform hardware throughput-monitor threshold *percentage* **interval** *seconds*

Example

```
Router# set platform hardware throughput-monitor threshold 85 interval 30
```

To display the platform hardware throughput monitor settings, use the **show platform hardware throughput-monitor parameters** command, as shown in the following example:

```
Router# show platform hardware throughput-monitor parameters

  Throughput monitor parameters
  Throughput monitor threshold: 95 percent
  Throughput monitor interval: 300 seconds
  Throughput monitor status: enabled
```

The following example shows a console log message received when the average throughput has exceeded 95 percent of the maximum throughput with a sample period of 300 seconds:

```
Dec 13 22:00:29.699: %BW_LICENSE-3-THROUGHPUT_THRESHOLD_LEVEL: F0: cpp_ha: Average
```

```
throughput rate exceeded 95 percent of licensed bandwidth 3 times, sample
 period 300 seconds, in last 24 hours
```

The following example shows a console log message received when the average throughput approaches maximum allowed throughput set by the installed license:

```
Dec 13 22:00:29.699: %BW_LICENSE-4-THROUGHPUT_MAX_LEVEL: F0: cpp_ha: Average
throughput rate exceeded the total licensed bandwidth 50000000 bps and dropped
packets 7 times, sample period 300 seconds, in last 24 Hours
```

To disable the platform hardware throughput monitor, perform the following command:

set platform hardware throughput-monitor disable

Requesting a New Virtual UDI

The router's license is node-locked to the vUDI. If you clone the router's VM to a new VM instance, the vUDI is in most cases automatically updated when the router first boots up on the cloned machine. However, if the vUDI is not automatically updated, you must manually request a new vUDI on the cloned VM instance.

Caution Requesting a new vUDI will invalidate the existing license. If you later need to rehost the license due to a system failure, you may need to perform additional steps on the Cisco Software Licensing portal. For more information on rehosting the router license, see Voluntarily Rehosting the License to a New VM, on page 409 and Obtaining a Rehost License if the System Fails, on page 410.

Perform the following step in EXEC mode:

SUMMARY STEPS

1. **request license new-udi**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**request license new-udi** **Example:** `Router# request license new-udi`	Requests that a new virtual UDI be assigned to the router's VM instance.

What to do next

Once you enter the **request license new-udi** command, you will be prompted to confirm, and then you will receive a series of system messages confirming the request:

```
Executing this command will invalidate the existing license,
proceed with generating new-udi?[confirm]
New udi CSR1000v:9MF19951DMU
```

```
Router#
*Aug 21 11:24:27.275:  found an eval license info: csr1kv_medium
*Aug 21 11:24:27.276:  Step 3. deletion of NOT-in-use licenses
*Aug 21 11:24:27.276:  Step 4. deletion of in-use licenses
*Aug 21 11:24:27.440: %LICENSE-2-UDI_CHANGED: UDI of this instance changed from OLD:
CSR1000V:9YA3086B993 to
New: CSR1000V:9MF19951DMU
```

To display the UDI history of the router's feature license, including previous virtual UDIs, enter the **show license udi history** command. The following example displays the UDI history of the feature license of a Cisco CSR 1000v:

```
Router# show license udi history
SlotID   PID                   SN                      UDI
------------------------------------------------------------------------------
*        CSR1000V              9MF19951DMU     CSR1000V:9MF19951DMU
 Invalidated UDIs:
------------------------------------------------------------
1. CSR1000V : 9YA3086B993
```

Cisco Software Licensing (IOS XE 3.12 or Earlier)

Activating CSL Evaluation Licenses for Cisco IOS XE 3.12S and Earlier

Note Licenses provided in Cisco IOS XE 3.12S and earlier (Standard, Advanced, and Premium) are no longer available. This material is provided as legacy information.

When the Cisco CSR 1000v first boots, the network interfaces are activated but feature support is limited and the throughput is limited to 2.5 Mbps. The evaluation license is bundled with the software, but you must activate the evaluation license to access the features.

The evaluation license expires 60 days from the time it is activated.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **license boot level {standard | advanced | premium}**
4. **end**
5. **write memory**
6. **reload**
7. show license detail
8. show platform hardware throughput level

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**license boot level {standard | advanced | premium}** **Example:** **Example:** **Example:** **Example:** `Router(config)# license boot level advanced`	Activates the evaluation license on the Cisco CSR 1000v upon the next reload. You must accept the End User License Agreement (EULA) to use the evaluation license. **Note** In Cisco IOS XE 3.12.1S and later 3.12.xS releases, use the **standard** option for the IPBase feature set, the **advanced** option for the Security feature set, and the **premium** option for the AX feature set.
Step 4	**end** **Example:** `Router(config)# end`	Exits global configuration mode.
Step 5	**write memory** **Example:** `Router# write memory`	Saves the running configuration to NVRAM.
Step 6	**reload** **Example:** `Router# reload`	Restarts the Cisco CSR 1000v to boot to the feature level set using the **license boot level** command.
Step 7	show license detail **Example:** `Router# show license detail`	After the Cisco CSR 1000v restarts, verifies that the license has been installed and is active.

	Command or Action	Purpose

Step 8	show platform hardware throughput level
	Example:
	`Router# show platform hardware throughput level`
	Example:
	`The current throughput level is 2500 kb/s`
		Verifies the Cisco CSR 1000v maximum throughput level.

What to do next

The evaluation license expires 60 days from the time it is activated.

Installing CSL Regular Licenses for Cisco IOS XE 3.12S and Earlier

Note Licenses provided in Cisco IOS XE 3.12S and earlier (Standard, Advanced, and Premium) are no longer available. This material is provided as legacy information.

In Cisco IOS XE 3.12S and earlier, the Cisco CSR 1000v first boots in limited mode with the Standard feature set enabled and the maximum throughput limited to 2.5 Mbps.

You can generate multiple licenses for the Cisco CSR 1000v from one PAK. The purchased PAK determines the number of licenses you can generate.

Repeat these steps for each license available for your PAK.

SUMMARY STEPS

1. Obtain the PAK.
2. **enable**
3. **show license udi**
4. Convert the PAK to a license by entering the PAK and the UDI into the Cisco Product License registration portal: "http:/www.cisco.com/go/license"
5. license install *stored-location-url*
6. **configure terminal**
7. **license boot level {standard | advanced** | **premium**}
8. **end**
9. **write memory**
10. **reload**
11. show license detail
12. **end**
13. **configure terminal**
14. **platform hardware throughput level MB**{**10 | 100 | 1000 | 250 | 2500 | 50 | 500 | 5000**}
15. **end**

16. show platform hardware throughput level

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	Obtain the PAK.	The PAK is provided to you when you order or purchase the right to use a feature set. • The PAK serves as a receipt and is used as part of the process to obtain a license.
Step 2	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 3	**show license udi** **Example:** `Router# show license udi`	Displays all the UDI values that can be licensed in a system. • You need the UDI of the device as part of the process to obtain a license.
Step 4	Convert the PAK to a license by entering the PAK and the UDI into the Cisco Product License registration portal: "http:/www.cisco.com/go/license" **Example:** `When entering the UDI, enter only the 11-character serial number, for example, 966975BITWG. The UDI is case-sensitive, and should be entered in all capital letters.`	After entering the appropriate information, you will receive an e-mail containing the license information that you can use to install the license: • Copy the license file received from the Cisco Product License Registration portal to the appropriate file system on the device.
Step 5	license install *stored-location-url* **Example:** `Router# license install bootflash:90NVHJ3C26E_20140724194119019.lic`	Installs the license. • Accept the end-user license agreement if prompted.
Step 6	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 7	**license boot level {standard | advanced | premium}** **Example:** `Router(config)# license boot level advanced`	Activates the license on the Cisco CSR 1000v upon the next reload. **Note** In Cisco IOS XE 3.12.1S and later 3.12.xS releases, use the **standard** option for the IPBase feature set, the

	Command or Action	Purpose
		advanced option for the Security feature set, and the **premium** option for the AX feature set.
Step 8	**end** **Example:** `Router(config)# end`	Exits global configuration mode.
Step 9	**write memory** **Example:** `Router# write memory`	Saves the running configuration to NVRAM.
Step 10	**reload** **Example:** `Router# reload`	Restarts the Cisco CSR 1000v to enable the feature set and the maximum throughput supported by the license.
Step 11	show license detail **Example:** `Router# show license detail`	After the Cisco CSR 1000v restarts, verifies that the license has been installed and is active.
Step 12	**end** **Example:** `Router(config)# end`	Exits global configuration mode.
Step 13	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 14	**platform hardware throughput level MB** {**10** | **100** | **1000** | **250** | **2500** | **50** | **500** | **5000**} **Example:** `Router(config)# platform hardware throughput level 500`	(Optional) Changes the maximum throughput level for the Cisco CSR 1000v. The **available throughput options vary depending on the release version.** **Note** Rebooting the Cisco CSR 1000v is not required.
Step 15	**end** **Example:** `Router(config)# end`	Exits global configuration mode.
Step 16	show platform hardware throughput level **Example:**	Verifies that the Cisco CSR 1000v maximum throughput level matches that of the installed license.

	Command or Action	Purpose

	`Router# show platform hardware throughput level`
	Example:
	`The current throughput level is 50000 kb/s`	

What to do next

Repeat these steps for each license available for your PAK.

The following is an example of the **show license detail** command showing an installed active license:

```
Router# show license detail
Index: 1        Feature: prem_100M                      Version: 1.0
        License Type: Permanent
        License State: Active, In Use
        License Count: Non-Counted
        License Priority: Medium
        Store Index: 0
        Store Name: Primary License Storage
```

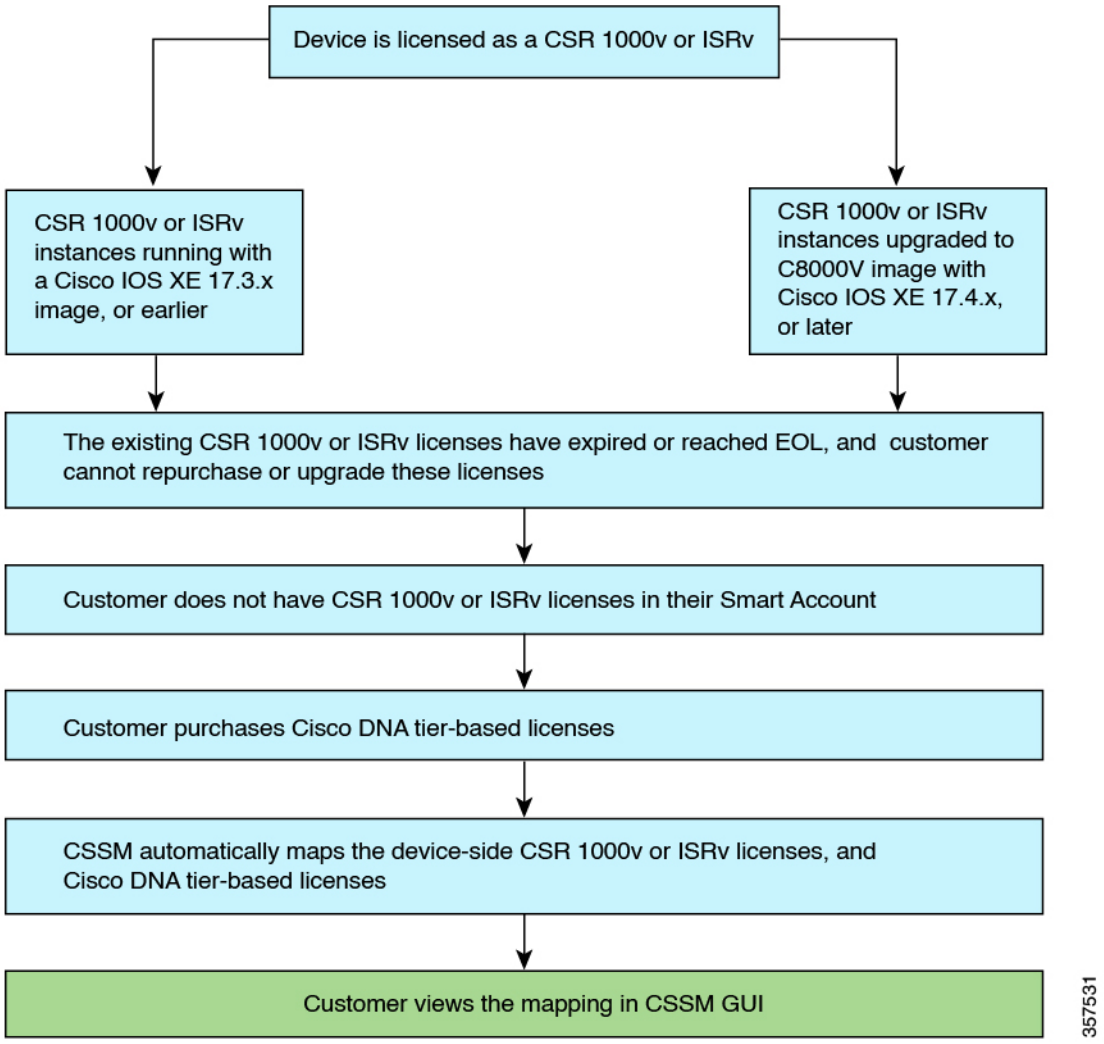
Automatic Mapping of Cisco CSR 1000v and Cisco ISRv Licenses to Cisco DNA Licenses

The Cisco Smart Software Manager (CSSM) manages the licenses for all Cisco devices, including Cisco CSR 1000v and Cisco ISRv. When these licenses reach end-of-life (EOL) in 2022, you must move to the Cisco DNA licenses to continue to use these devices, and to keep your Smart Account compliant. After you activate your Cisco DNA license, the CSSM automatically maps the device licenses (CSR 1000v and Cisco ISRv APPX, AX, SEC and IP Base licenses) to your Cisco DNA license, based on the existing license package and throughput.

Prerequisites for Automatic Mapping of Cisco CSR 1000v and Cisco ISRv Licenses to Cisco DNA Licenses

The CSSM maps the device licenses and Cisco DNA licenses when the following set of conditions are met:

Figure 7: Prerequisites for Automatic Mapping of Cisco CSR 1000v and Cisco ISRv Licenses to Cisco DNA Licenses



Note From Cisco IOS XE 17.4, all platforms support only Smart Licensing using Policy. For Cisco CSR 1000v and Cisco ISRv platforms on Cisco IOS XE 17.3.x or earlier, you must upgrade to Cisco IOS XE 17.4 to move from Smart Licensing to Smart Licensing using Policy.

For details on Cisco CSR 1000v and Cisco ISRv upgrade to Cisco C8000V, see the Cisco Catalyst 8000V Edge Software Installation and Configuration Guide.

For details on Smart Licensing Using Policy, see Smart Licensing Using Policy for Cisco Enterprise Routing Platforms.

Information About Automatic Mapping of Cisco CSR 1000v and Cisco ISRv Licenses to Cisco DNA Licenses

The CSSM maps Cisco CSR 1000v and Cisco ISRv device-side licenses to Cisco DNA licenses based on package type and throughput level.

The IP Base and Security licenses are mapped to the Network Essentials package, and the APPX and AX licenses are mapped to the Network Advantage package.

License throughputs from 10 Mbps to 10 Gbps are mapped to tiers 0 to 3.

Table 32: License Mapping Based on Package

CSR 1000v or Cisco ISRv License Package	Cisco DNA License Package
IP Base, Security	Network Essentials
APPX, AX	Network Advantage

Table 33: License Mapping Based on Throughput

Throughput	Tier
10 Mbps	Tier 0 (T0)
50 Mbps, 100 Mbps	Tier 1 (T1)
250 Mbps, 500 Mbps, 1 Gbps	Tier 2 (T2)
2.5 Gbps, 5 Gbps, 10 Gbps	Tier 3 (T3)

Examples: Automatic Mapping of Cisco CSR 1000v and Cisco ISRv Licenses to Cisco DNA Licenses

Example 1: Cisco CSR 1000v AX 10 Mbps License Mapped to Cisco DNA Network Advantage Tier 0 License

In this example, the customer's device device has a Cisco CSR 1000v AX and 10 Mbps throughput license. When the customer purchases a Cisco DNA license, the CSSM maps this as a Network Advantage Tier 0 license.

Package type: AX = Network Advantage

Throughput: 10 Mbps = Tier 0

You can view the license mapping in CSSM GUI.

Figure 8: CSSM View - CSR 1000v AX 10 Mbps License Mapped to Network Advantage Tier 0

Figure 9: CSSM View - CSR 1000v AX 10 Mbps License Mapped to Network Advantage Tier 0

Example 2: Cisco ISRv SEC 1 Gbps License Mapped to Cisco DNA Network Essentials Tier 2 License

In this example, the customer's device has a Cisco ISRv SEC and 1 Gbps throughput license. When the customer purchases a Cisco DNA license, the CSSM maps this as a Network Essentials Tier 2 license.

Package type: SEC = Network Essentials

Throughput: 1 Gbps = Tier 2

You can view the license mapping in CSSM GUI.

Figure 10: CSSM View - ISRv SEC 1 Gbps License Mapped to Network Essentials Tier 2

Figure 11: CSSM View - ISRv SEC 1 Gbps License Mapped to Network Essentials Tier 2

Device and Network Stack Licenses

The following tables provide information about the corresponding network stack license for every device license and throughput combination.

Table 34: Cisco CSR 1000v Network Stack Licenses

Device License	Throughput	Tier	Network Stack License
CSR-10M-IPB CSR-10M-SEC	10 Mbps	T0	NWSTACK-T0-E
CSR-50M-IPB CSR-100M-IPB CSR-50M-SEC CSR-100M-SEC	50 Mbps 100 Mbps	T1	NWSTACK-T1-E
CSR-250M-IPB CSR-500M-IPB CSR-1G-IPB CSR-250M-SEC CSR-500M-SEC CSR-1G-SEC	250 Mbps 500 Mbps 1 Gbps	T2	NWSTACK-T2-E
CSR-2.5G-IPB CSR-5G-IPB CSR-10G-IPB CSR-2.5G-SEC CSR-5G-SEC CSR-10G-SEC	2.5 Gbps 5 Gbps 10 Gbps	T3	NWSTACK-T3-E
CSR-10M-APPX CSR-10M-AX	10 Mbps	T0	NWSTACK-T0-A
CSR-50M-APPX CSR-100M-APPX CSR-50M-AX CSR-100M-AX	50 Mbps 100 Mbps	T1	NWSTACK-T1-A

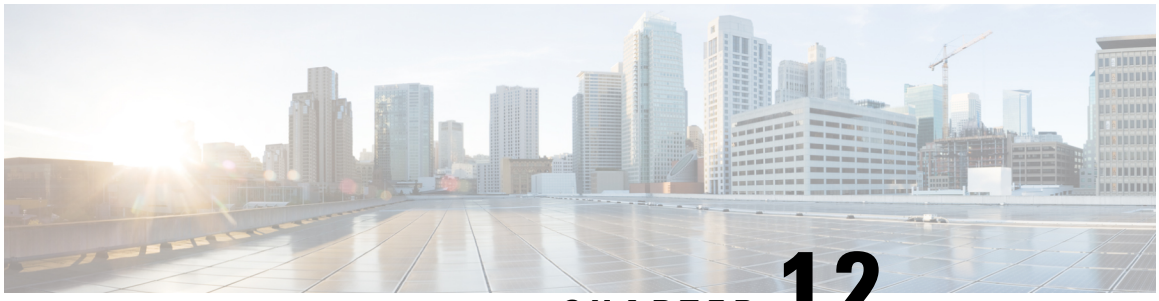
Device License	Throughput	Tier	Network Stack License
CSR-250M-APPX CSR-500M-APPX CSR-1G-APPX CSR-250M-AX CSR-500M-AX CSR-1G-AX	250 Mbps 500 Mbps 1 Gbps	T2	NWSTACK-T2-A
CSR-2.5G-APPX CSR-5G-APPX CSR-10G-APPX CSR-2.5G-AX CSR-5G-AX CSR-10G-AX	2.5 Gbps 5 Gbps 10 Gbps	T3	NWSTACK-T3-A

Table 35: Cisco ISRv Network Stack Licenses

Device License	Throughput	Tier	Network Stack License
ISR-10M-IPB ISR-10M-SEC	10 Mbps	T0	NWSTACK-T0-E
ISR-50M-IPB ISR-100M-IPB ISR-50M-SEC ISR-100M-SEC	50 Mbps 100 Mbps	T1	NWSTACK-T1-E
ISR-250M-IPB ISR-500M-IPB ISR-1G-IPB ISR-250M-SEC ISR-500M-SEC ISR-1G-SEC	250 Mbps 500 Mbps 1 Gbps	T2	NWSTACK-T2-E

Device License	Throughput	Tier	Network Stack License
ISR-2.5G-IPB ISR-5G-IPB ISR-10G-IPB ISR-2.5G-SEC ISR-5G-SEC ISR-10G-SEC	2.5 Gbps 5 Gbps 10 Gbps	T3	NWSTACK-T3-E
ISR-10M-APPX ISR-10M-AX	10 Mbps	T0	NWSTACK-T0-A
ISR-50M-APPX ISR-100M-APPX ISR-50M-AX ISR-100M-AX	50 Mbps 100 Mbps	T1	NWSTACK-T1-A
ISR-250M-APPX ISR-500M-APPX ISR-1G-APPX ISR-250M-AX ISR-500M-AX ISR-1G-AX	250 Mbps 500 Mbps 1 Gbps	T2	NWSTACK-T2-A
ISR-2.5G-APPX ISR-2.5G-AX	2.5 Gbps	T3	NWSTACK-T3-A

Note Cisco ISRv devices do not support 5 Gbps and 10 Gbps throughputs.



CHAPTER 12

Upgrading the Cisco IOS XE Software

Prerequisites for the Software Upgrade Process

This section describes how to upgrade the Cisco IOS XE software for an existing Cisco CSR 1000v or Cisco ISRv installation on a VM. For information on installing a new Cisco CSR 1000v, see Cisco CSR 1000v Series Cloud Services Router Overview, on page 5.

This procedure is for upgrading to a new software version on the same VM only. It does not describe how to install or rehost an existing router running the same or upgraded software version on a different VM.

Note The router does not support In-Service Software Upgrade (ISSU).

Note (Cisco IOS XE Everest 16.5 and later) On Amazon Web Services (AWS), you can use the Cisco CSR 1000v .bin file to upgrade the version of the router, without having to recreate the AWS EC2 instance from a new AMI.

Note (Cisco IOS XE Everest 16.4 and earlier) On Amazon Web Services (AWS), you cannot use the Cisco CSR 1000v .bin file to upgrade AMIs obtained from AWS. You must create a new AMI instance and migrate your configuration and licenses.

Note (Cisco IOS XE Fuji 16.7 or later) On Microsoft Azure, to do an in-place upgrade of the Cisco CSR 1000v .bin file you must follow the steps in Upgrading a Cisco IOS XE Image on Microsoft Azure. This is within the Configuring Cisco CSR 1000v on Microsoft Azure section of the Cisco CSR 1000v Deployment Guide for Microsoft Azure.

Note (Cisco IOS XE Everest 16.6 and earlier) On Microsoft Azure, you cannot use the Cisco CSR 1000v .bin file to upgrade a Cisco CSR1000v instance. You must re-deploy a new instance from the Microsoft Azure Portal and migrate your configuration and licenses.

Caution If upgrading to Cisco IOS XE Release 3.11S from an earlier release, we recommend that you update your configuration to remove the GigabitEthernet0 management interface before upgrading. Because the GigabitEthernet0 interface is no longer supported beginning with Cisco IOS XE Release 3.11S, you will receive system errors if the upgraded configuration includes this interface.If downgrading from Cisco IOS XE Release 3.11S to an earlier release, note also that the management interface will need to change to GigabitEthernet0 for the earlier release.

Be sure to complete the following prerequisites for upgrading the Cisco IOS XE version of the router software image:

Read the Cisco CSR 1000v Series Cloud Services Router Release Notes to verify the following:

- This is a note about the compatibility between the hypervisor vendor and Cisco IOS XE version. If you want to upgrade to a new hypervisor version not supported by your current version of Cisco IOS XE on the Cisco CSR 1000v/ISRv, you need to upgrade the version before upgrading to the new hypervisor version.
- System requirements for the x86 hardware that may differ from those of the currently running on the router.
- Memory requirements of the VM for the Cisco CSR 1000v/ISRv software image.

Note If the new router's version requires more memory than your previous version, you must increase the memory allocation on the VM before beginning the upgrade process.

- Software features supported on the upgraded Cisco IOS XE version.
- Any upgrade restrictions.

Obtain the Cisco CSR 1000v/ISRv software image from Cisco.com. For the Cisco CSR 1000v, see Obtaining the Cisco CSR 1000v VM Image, on page 65 .

Note You must use the .bin file to upgrade or downgrade your software. The .iso and .ova files are used for first-time installation only.

Saving Backup Copies of Your Old System Image and Configuration

To avoid unexpected downtime in the event you encounter serious problems using a new system image or startup configuration, we recommend that you save backup copies of your current startup configuration file and Cisco IOS XE software system image file on a server.

For more detailed information, see the "Managing Configuration Files" chapter in the Managing Configuration Files Configuration Guide, Cisco IOS XE Everest 16.6.

To save backup copies of the startup configuration file and the system image file, complete the following steps.

SUMMARY STEPS

1. **enable**
2. **copy nvram:startup-config** {**ftp:** | **rcp:** | **tftp:**}
3. **dir bootflash:**
4. **copy bootflash:** {**ftp:** | **rcp:** | **tftp:**}

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**copy nvram:startup-config** {**ftp:** | **rcp:** | **tftp:**} **Example:** `Router# copy nvram:startup-config ftp:`	Copies the startup configuration file to a server. • The configuration file copy can serve as a backup copy. • Enter the destination URL when prompted.
Step 3	**dir bootflash:** **Example:** `Router# dir bootflash:`	Displays the layout and contents of a bootflash memory file system. **bootflash:** is aliased onto **flash:**. • Learn the name of the system image file.
Step 4	**copy bootflash:** {**ftp:** | **rcp:** | **tftp:**}	Copies a file from bootflash memory to a server.

	Command or Action	Purpose
	Example: `Router# copy bootflash: ftp:`	• Copy the system image file to a server. This file can serve as a backup copy. • Enter the bootflash memory partition number if prompted. • Enter the filename and destination URL when prompted.

What to do next

The following example shows the startup configuration being copied to a TFTP server:

```
Router# copy nvram:startup-config tftp:

Remote host[]? 192.0.2.1

Name of configuration file to write [rtr2-confg]? rtr2-config-b4upgrade

Write file rtr2-confg-b4upgrade on host 192.0.0.1?[confirm] <cr>

![OK]
```

The following example uses the **dir bootflash:** command in privileged EXEC mode to learn the name of the system image file and the **copy bootflash: tftp:** command in privileged EXEC mode to copy the system image to a TFTP server. The router uses the default username and password.

```
Router#
Router# dir bootflash:
Directory of bootflash:/
    1  -rw-    48311224   Mar 2 1901 11:32:50 +00:00 csr1000v-universalk9-mz.SSA.XFR_20090407

    2 -rw-         983  Feb 14 2021 12:41:52 +00:00  running-config
260173824 bytes total (211668992 bytes free)
Router# copy bootflash: tftp:
Source filename [running-config]?
Address or name of remote host []? 192.0.2.1
Destination filename [router-confg]? running-config
983 bytes copied in 0.048 secs (20479 bytes/sec)
Router#
```

Using TFTP or Remote Copy Protocol to Copy the System Image into Boot Flash Memory

The following details the logistics of upgrading the system image:

Install a TFTP server or an RCP server application on a TCP/IP-ready workstation or PC. Many third-party vendors provide free TFTP server software, which you can find by searching for “TFTP server” in a web search engine.

If you use TFTP:

- Configure the TFTP application to operate as a TFTP server, not a TFTP client.

- Specify the outbound file directory to which you will download and store the system image.
- Download the new Cisco IOS XE software image into the workstation or PC.
- Verify that the TFTP or RCP server has IP connectivity to the router. If you cannot successfully ping between the TFTP or RCP server and the router, either configure a default gateway on the router or make sure that the router and server each have an IP address in the same network or subnet.

SUMMARY STEPS

1. **enable**
2. Use one of the following commands to copy a file from a server to bootflash memory:
   - **copy tftp bootflash:**
   - **copy rcp bootflash**
3. When prompted, enter the IP address of the TFTP or RCP server:
4. When prompted, enter the filename of the Cisco IOS software image to be installed:
5. When prompted, enter the filename as you want it to appear on the router. Typically, the same filename is entered as was used in the previous step.
6. If an error message appears that says, "Not enough space on device," do the following:
7. If the error message does not appear, enter **no** when prompted to erase the bootflash memory before copying:

DETAILED STEPS

Procedure

Step 1 **enable**

Use this command to enter privileged EXEC mode. Enter your password if prompted:

Example:

```
Router> enable
Password: <password>
Router#
```

Step 2 Use one of the following commands to copy a file from a server to bootflash memory:

- **copy tftp bootflash:**
- **copy rcp bootflash**

Example:

```
Router# copy tftp bootflash:
```

Step 3 When prompted, enter the IP address of the TFTP or RCP server:

Example:

```
Address or name of remote host []? 10.10.10.2
```

Step 4 When prompted, enter the filename of the Cisco IOS software image to be installed:

Example:

```
Source filename ? csr1000v-universalk9.03.10.00.S.153-3.S-ext.SPA.bin
```

Note
The filename is case sensitive.

Step 5 When prompted, enter the filename as you want it to appear on the router. Typically, the same filename is entered as was used in the previous step.

Example:

```
Destination filename ? csr1000v-universalk9.03.10.00.S.153-3.S-ext.SPA.bin
```

Step 6 If an error message appears that says, "Not enough space on device," do the following:

- If you are certain that all the files in bootflash memory should be erased, enter **y** when prompted twice to confirm that bootflash memory will be erased before copying:

Example:

```
Accessing tftp://10.10.10.2/csr1000v-universalk9.03.10.00.S.153-3.S-ext.SPA.bin...
     Erase bootflash: before copying? [confirm] y
     Erasing the flash filesystem will remove all files! Continue? [confirm] y

     Erasing device...
```

- If you are not certain that all files in bootflash memory should be erased, press **Ctrl-Z**.

Step 7 If the error message does not appear, enter **no** when prompted to erase the bootflash memory before copying:

Example:

```
Accessing tftp://10.10.10.2/csr1000v-universalk9.03.10.00.S.153-3.S-ext.SPA.bin...
     Erase bootflash: before copying? [confirm] no
```

Loading the New System Image from the Cisco IOS XE Software

SUMMARY STEPS

1. **dir bootflash:**
2. **configure terminal**
3. **no boot system**
4. **boot system bootflash:***system-image-filename*.bin
5. (Optional) Repeat the previous step to specify the order in which the router should attempt to load any backup system images.
6. **exit**

7. **write**
8. **show version**
9. If the last digit in the configuration register is 0 or 1, proceed to the next step. However, if the last digit in the configuration register is between 2 and F, proceed to the step "copy running-config startup-config" below.
10. **configure terminal**
11. **config-register 0x2102**
12. **exit**
13. **copy running-config startup-config**
14. **write memory**
15. **reload**
16. When prompted to save the system configuration, enter **no**:
17. When prompted to confirm the reload, enter **y**:
18. **show version**

DETAILED STEPS

Procedure

Step 1 **dir bootflash:**

Use this command to display a list of all files and directories in bootflash memory:

Example:

```
Router# dir bootflash:

Directory of bootflash:/
    3  -rw-     6458388   Mar 01 1993 00:00:58 csr1000v.tmp
 1580  -rw-     6462268   Mar 06 1993 06:14:02 csr1000v-ata
63930368 bytes total (51007488 bytes free)
```

Step 2 **configure terminal**

Use this command to enter global configuration mode:

Example:

```
Router# configure terminal
Router(config)#
```

Step 3 **no boot system**

Use this command to delete all entries in the bootable image list, which specifies the order in which the router attempts to load the system images at the next system reload or power cycle:

Example:

```
Router(config)# no boot system
```

Step 4 **boot system bootflash:***system-image-filename*.bin

Note

If the new system image is the first file or the only file displayed in the **dir bootflash:** command output in Step 1, you do not need to perform this step.

Use this command to load the new system image after the next system reload or power cycle. For example:

Example:

```
Router(config)# boot system bootflash:
csr1000v-universalk9.03.10.00.S.153-3.S-ext.SPA.bin
```

Step 5 (Optional) Repeat the previous step to specify the order in which the router should attempt to load any backup system images.

Step 6 **exit**

Use this command to exit global configuration mode:

Example:

```
Router(config)# exit
 Router#
```

Step 7 **write**

or

write memory

Example:

```
Router# write memory
```

Note
This step is required beginning with Cisco IOS XE Release 3.9S if upgrading to a later version. Entering the **write** or **write memory** command updates the GRUB menu list of images available on the bootflash disk.

Step 8 **show version**

Use this command to display the configuration register setting:

Example:

```
Router# show version

Cisco Internetwork Operating System Software
.
.
.
Configuration register is 0x0

Router#
```

Step 9 If the last digit in the configuration register is 0 or 1, proceed to the next step. However, if the last digit in the configuration register is between 2 and F, proceed to the step "copy running-config startup-config" below.

Step 10 **configure terminal**

Use this command to enter global configuration mode:

Example:

```
Router# configure terminal
Router(config)#
```

Step 11 **config-register 0x2102**

Use this command to set the configuration register so that, after the next system reload or power cycle, the router loads a system image from the **boot system** commands in the startup configuration file:

Example:

```
Router(config)# config-register 0x2102
```

Note
The 0x2102 value is the default configuration register setting. If you didn't change this setting from the default, this step is not required.

Step 12 **exit**

Use this command to exit global configuration mode:

Example:

```
Router(config)# exit
 Router#
```

Step 13 **copy running-config startup-config**

Use this command to copy the running configuration to the startup configuration:

Example:

```
Router# copy running-config startup-config
```

Step 14 **write memory**

Note
This step is required beginning with Cisco IOS XE Release 3.9S if upgrading to a later version. Entering the **write memory** command updates the GRUB menu list of images available on the bootflash disk.

Step 15 **reload**

Use this command to reload the operating system:

Example:

```
Router# reload
```

Step 16 When prompted to save the system configuration, enter **no**:

Example:

```
System configuration has been modified. Save? [yes/no]: no
```

Step 17 When prompted to confirm the reload, enter **y**:

Example:

```
Proceed with reload? [confirm] y
```

Step 18 **show version**

Use this command to verify that the router loaded the proper system image:

Example:

```
Router# show version
00:22:25: %SYS-5-CONFIG_I: Configured from console by console
Cisco Internetwork Operating System Software
.
.
.
System returned to ROM by reload
System image file is "bootflash:
csr1000v-universalk9.03.10.00.S.153-3.S-ext.SPA.bin"
```

Loading the New System Image from GRUB Mode

To load the new system image from GRand Unified Bootloader (GRUB) mode, follow these steps, beginning in EXEC mode.

SUMMARY STEPS

1. **dir bootflash:**
2. **configure terminal**
3. **boot system bootflash:***system-image-filename*.bin
4. **do write**
5. **config-register 0x0000**
6. At the **grub>** prompt, enter ESC to access the GRUB menu.
7. Select the .bin file to upgrade the software image on the router to the new version.
8. Press **Enter** to boot the selected image to begin the upgrade process.

DETAILED STEPS

Procedure

Step 1 **dir bootflash:**

Use this command to display a list of all files and directories in bootflash memory:

Example:

```
Router# dir bootflash:

Directory of bootflash:/
    3  -rw-     6458388   Mar 01 1993 00:00:58  csr1000v.tmp
 1580  -rw-     6462268   Mar 06 1993 06:14:02 csr1000v-ata
63930368 bytes total (51007488 bytes free)
```

Step 2 **configure terminal**

Use this command to enter global configuration mode:

Example:

```
Router# configure terminal
Router(config)#
```

Step 3 **boot system bootflash:***system-image-filename*.bin

Note
If the new system image is the first file or the only file displayed in the **dir bootflash:** command output , you do not need to perform this step.

Use this command to load the new system image after the next system reload or power cycle. For example:

Example:

```
Router(config)# boot system bootflash:
csr1000v-universalk9.03.10.00.S.153-3.S-ext.SPA.bin
```

Step 4 **do write**

or

do write memory

Example:

```
Router(config)# do write memory
```

Note
This step is required beginning with Cisco IOS XE Release 3.9S if upgrading to a later version. Entering the **do write** or **do write memory** command updates the GRUB menu list of images available on the bootflash disk.

Step 5 **config-register 0x0000**

Use this command to enter GRUB mode.

The following shows an example of entering GRUB mode.

Example:

```
Router(config)# config-register 0x0000

GNU GRUB  version 0.97  (638K lower / 3143616K upper memory)
 [ Minimal BASH-like line editing is supported.  For the first word, TAB
   lists possible command completions.  Anywhere else TAB lists the possible
   completions of a device/filename.  ESC at any time exits to menu. ]
grub> help
 [ Minimal BASH-like line editing is supported.  For the first word, TAB
   lists possible command completions.  Anywhere else TAB lists the possible
   completions of a device/filename.  ESC at any time exits to menu. ]
confreg [VALUE]                       help [--all] [PATTERN ...]
grub>
```

Step 6 At the **grub>** prompt, enter ESC to access the GRUB menu.

The GRUB menu displays, showing the images that are available to boot.

Example:

```
GNU GRUB  version 0.97  (638K lower / 3143616K upper memory)
```

```
+--------------------------------------------------------------------------+
| CSR1000v - csr1000v-universalk9.03.10.00.S.153-3.S-ext.SPA.bin           |
| CSR1000v - packages.conf                                                 |
| CSR1000v - GOLDEN IMAGE                                                  |
|                                                                          |
|                                                                          |
|                                                                          |
|                                                                          |
|                                                                          |
|                                                                          |
|                                                                          |
|                                                                          |
+--------------------------------------------------------------------------+
     Use the ^ and v keys to select which entry is highlighted.
     Press enter to boot the selected OS, or 'c' for a command-line.
```

Select the image to boot the router from using the up and down arrow key. To return to the GRUB prompt, enter the letter **c**.

Step 7 Select the .bin file to upgrade the software image on the router to the new version.

Step 8 Press **Enter** to boot the selected image to begin the upgrade process.

Saving Backup Copies of Your New System Image and Configuration

To aid file recovery and to minimize downtime in the event of file corruption, we recommend that you save backup copies of the startup configuration file and the Cisco IOS software system image file on a server.

Tip Do not erase any existing backup copies of your configuration and system image that you saved before upgrading your system image. If you encounter serious problems using your new system image or startup configuration, you can quickly revert to the previous working configuration and system image.

To save backup copies of the startup configuration file and the system image file, complete the following steps.

SUMMARY STEPS

1. **enable**
2. **copy nvram:startup-config** {**ftp:** | **rcp:** | **tftp:**}
3. **dir bootflash:**
4. **copy bootflash:** {**ftp:** | **rcp:** | **tftp:**}

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**copy nvram:startup-config** {**ftp:** | **rcp:** | **tftp:**} **Example:** `Router# copy nvram:startup-config ftp:`	Copies the startup configuration file to a server. • The configuration file copy serves as a backup copy. • Enter the destination URL when prompted.
Step 3	**dir bootflash:** **Example:** `Router# dir bootflash:`	Displays the layout and contents of a bootflash memory file system. • Write down the name of the system image file.
Step 4	**copy bootflash:** {**ftp:** | **rcp:** | **tftp:**} **Example:** `Router# copy bootflash: ftp:`	Copies a file from bootflash memory to a server. • Copy the system image file to a server to serve as a backup copy. • Enter the bootflash memory partition number if prompted. • Enter the filename and destination URL when prompted.

What to do next

The following example shows the startup configuration being copied to a TFTP server:

```
Router# copy nvram:startup-config tftp:

Remote host[]? 172.16.101.101

Name of configuration file to write [rtr2-confg]? <cr>

Write file rtr2-confg on host 172.16.101.101?[confirm] <cr>

![OK]
```

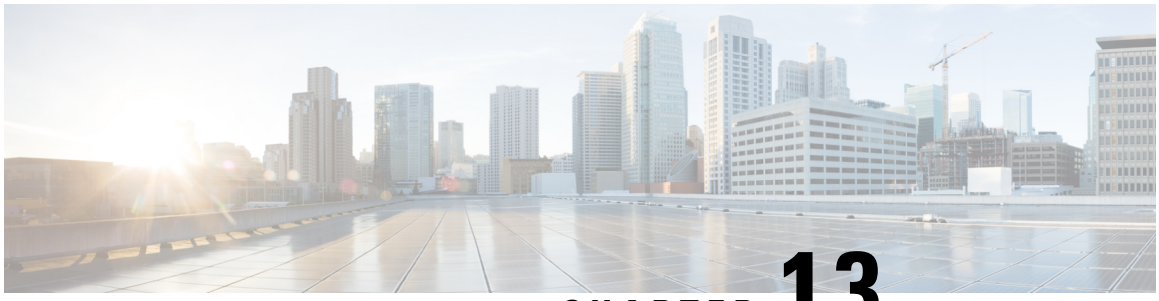
The following example uses the **dir bootflash:** privileged EXEC command to obtain the name of the system image file and the **copy bootflash: tftp:** privileged EXEC command to copy the system image to a TFTP server. The router uses the default username and password.

```
Router# dir bootflash:

System flash directory:
```

```
File Length Name/status
1 4137888 csr1000v-universalk9.03.10.00.S.153-3.S-ext.SPA.bin
[4137952 bytes used, 12639264 available, 16777216 total]
16384K bytes of processor board System flash (Read/Write)\
Router# copy bootflash: tftp:

IP address of remote host [255.255.255.255]? 192.0.2.1
filename to write on tftp host? csr1000v-advernterprisek9-mz

writing csr1000v-universalk9.03.10.00.S.153-3.S-ext.SPA!!!!...
successful ftp write.
```

Rebooting the Cisco CSR 1000v

Once you have copied the new system image into bootflash memory, loaded the new system image and saved a backup copy of the new system image and configuration, you need to reboot the VM. See your VM vendor documentation for more information about rebooting the VM. After rebooting, the router VM should include the new system image with a newly installed version of the Cisco IOS XE software.



CHAPTER 13

Mapping Cisco CSR 1000v Network Interfaces to VM Network Interfaces

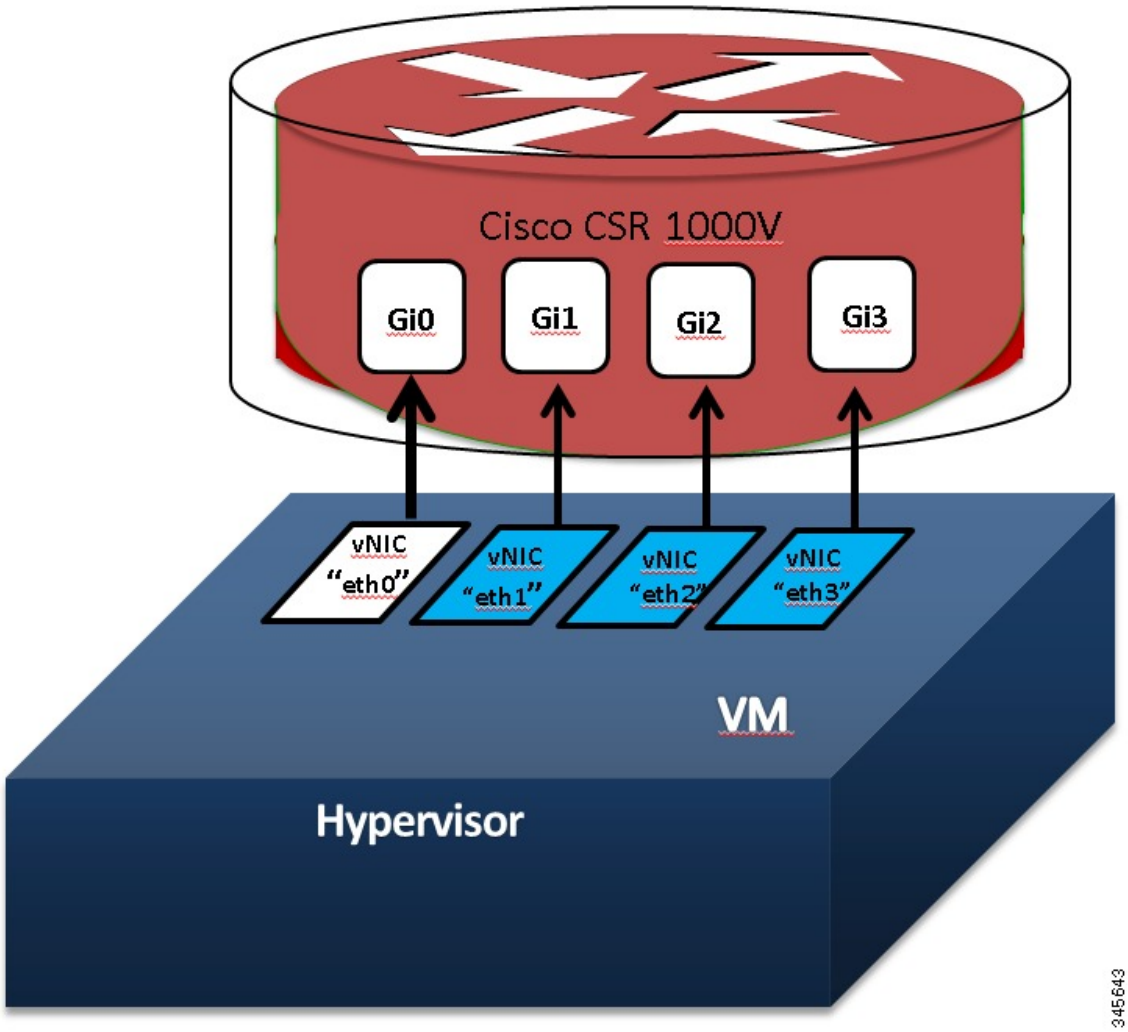
Mapping Cisco CSR 1000v Network Interfaces to VM Network Interfaces

Mapping the Router Network Interfaces to vNICs

The Cisco CSR 1000v maps the GigabitEthernet network interfaces to the logical virtual network interface card (vNIC) name assigned by the VM. The VM in turn maps the logical vNIC name to a physical MAC address.

When the Cisco CSR 1000v is booted for the first time, the router interfaces are mapped to the logical vNIC interfaces that were added when the VM was created. The figure below shows the relationship between the vNICs and the Cisco CSR 1000v router interfaces.

Figure 12: vNICs Mapped to Cisco CSR 1000v Router Interfaces



After the Cisco CSR 1000v boots, you need to display the mapping between the logical interface on the router with the vNIC and the vNIC MAC address using the **show platform software vnic-if interface-mapping** command. The display for this command is different depending on your Cisco IOS XE release version. (Note: For Cisco IOS XE Release 3.11 or later, the GigabitEthernet0 interface is no longer supported.)

```
Router# show platform software vnic-if interface-mapping
------------------------------------------------------------------------
Interface Name        Short Name     vNIC Name                 Mac Addr
------------------------------------------------------------------------
GigabitEthernet0       Gi0        eth0 (vmxnet3)           000c.2946.3f4d
GigabitEthernet2       Gi2        eth2 (vmxnet3)           0050.5689.0034
GigabitEthernet1       Gi1        eth1 (vmxnet3)           0050.5689.000b
------------------------------------------------------------------------
```

The vNIC name shown in the display is a logical interface that the Cisco CSR 1000v uses to map to the interface on the hypervisor. It does not always map to the corresponding NIC name added during the VM installation. For example, the logical “eth1” vNIC name in the display may not necessarily map to “NIC1” that was added in the VM installation process.

Caution It is important that you verify the interface mapping before you begin configuring the Gigabit Ethernet network interfaces on the Cisco CSR 1000v. This ensures that the network interface configuration will apply to the correct physical MAC address interface on the VM host.

If you reboot the router and do not add or delete any vNICs, the interface mapping remains the same as before. If you reboot the router and delete vNICs, special care must be taken to ensure that the configuration for the remaining interfaces remains intact. For more information, see Adding and Deleting Network Interfaces on the Cisco CSR 1000v, on page 253.

Note In Cisco IOS XE Release 3.10S and earlier, the first vNIC added is automatically mapped to the GigabitEthernet0 management interface. All subsequent vNICs added are mapped to router interfaces. Support for the GigabitEthernet0 interface was removed in Cisco IOS XE Release 3.11S.

Adding and Deleting Network Interfaces on the Cisco CSR 1000v

The Cisco CSR 1000v maps the router GigabitEthernet interfaces to the logical vNIC name assigned by the VM, which in turn is mapped to a MAC address on the VM host. You can add or delete vNICs on the VM to add or delete GigabitEthernet interfaces on the Cisco CSR 1000v. You can add vNICs while the router is active.

To delete a vNIC from the VM, you must first power down the VM. If you delete any vNICs, the router must be rebooted. For more information about adding and deleting vNICs, see the VMware Documentation .

Caution If you remove a vNIC without first updating the Cisco CSR 1000v network interface configuration, you risk a configuration mismatch when the router reboots. When the router reboots and a vNIC has been removed, the remaining logical vNIC names could get reassigned to different MAC addresses. As a result, the GigabitEthernet network interfaces on the Cisco CSR 1000v can be reassigned to different physical interfaces on the hypervisor.

Before you add or delete network interfaces, first verify the interface-to-vNIC mapping using the **show platform software vnic-if interface-mapping** command.

```
csr1000v# show platform software vnic-if interface-mapping
------------------------------------------------------------------------
Interface Name             Driver Name               Mac Addr
------------------------------------------------------------------------
GigabitEthernet3            vmxnet3                 000c.2946.3f4d
GigabitEthernet2            vmxnet3                 0050.5689.0034
GigabitEthernet1            vmxnet3                 0050.5689.000b
GigabitEthernet0            vmxnet3                 000c.2946.3f4d
------------------------------------------------------------------------
```

After adding or deleting network interfaces on the VM, always verify the new interface-to-vNIC mapping before making configuration changes to the network interfaces. The following example shows the interface mapping after a new vNIC has been added. The new vNIC maps to the GigabitEthernet4 network interface on the Cisco CSR 1000v.

```
csr1000v# show platform software vnic-if interface-mapping
------------------------------------------------------------------------
Interface Name             Driver Name               Mac Addr
------------------------------------------------------------------------
GigabitEthernet4            vmxnet3                 0010.0d40.37ff
GigabitEthernet3            vmxnet3                 000c.2946.3f4d
GigabitEthernet2            vmxnet3                 0050.5689.0034
GigabitEthernet1            vmxnet3                 0050.5689.000b
GigabitEthernet0            vmxnet3                 000c.2946.3f4d
------------------------------------------------------------------------
```

Removing a vNIC from a Running VM

To remove a vNIC from a running VM, use the `clear platform software` command (described below). Perform this command before removing a vNIC from the hypervisor configuration. This is part of a "two-step hot remove". To see which hypervisors support a two-step hot remove, look for hypervisors with vNIC Two-Step Hot Remove Support = Yes in Hypervisor Support, on page 8.

clear platform software vnic-if interface GigabitEthernet*interface-number*

interface-number—value from 0–32.

Example:

```
csr1000v# clear platform software vnic-if interface GigabitEthernet4
```

Next, remove the vNIC from the hypervisor configuration.

Note Starting with the IOS XE 17.1 release, you no longer need to execute the `clear platform software vnic-int interface` command before you remove the vNIC configuration from the hypervisor. This command will be deprecated in a future release.

Cisco CSR 1000v Network Interfaces and VM Cloning

When first installed, the Cisco CSR 1000v creates a database that maps the vNIC name to the MAC address. This database is used to maintain a persistent mapping between the router interfaces and the vNIC-to-MAC address mapping in case vNICs are added or deleted. The interfaces are mapped to the stored Universal Unique Identification (UUID) maintained by VMware.

The mapping between the router network interfaces and the vNICs only applies to the current VM that the Cisco CSR 1000v is installed on. If the VM is cloned, then the stored UUID will not match the current UUID and the interface mapping will not match the router configuration.

To prevent the interface mapping from becoming mis-matched, you need to perform the following steps on the original VM before cloning:

SUMMARY STEPS

1. Make sure the original VM includes the number of configured vNICs required on the cloned VM before beginning the cloning process.
2. Enter the **clear platform software vnic-if nvtable** command on the original VM.
3. Reboot the Cisco CSR 1000v.

4. On the cloned VM, verify the interface mapping using the **show platform software vnic-if interface-mapping** command.
5. Configure the router interfaces on the cloned VM accordingly.

DETAILED STEPS

Procedure

Step 1 Make sure the original VM includes the number of configured vNICs required on the cloned VM before beginning the cloning process.

Step 2 Enter the **clear platform software vnic-if nvtable** command on the original VM.

This command clears the persistent interface database on the original VM and updates the interface mapping to the hypervisor.

Step 3 Reboot the Cisco CSR 1000v.

Step 4 On the cloned VM, verify the interface mapping using the **show platform software vnic-if interface-mapping** command.

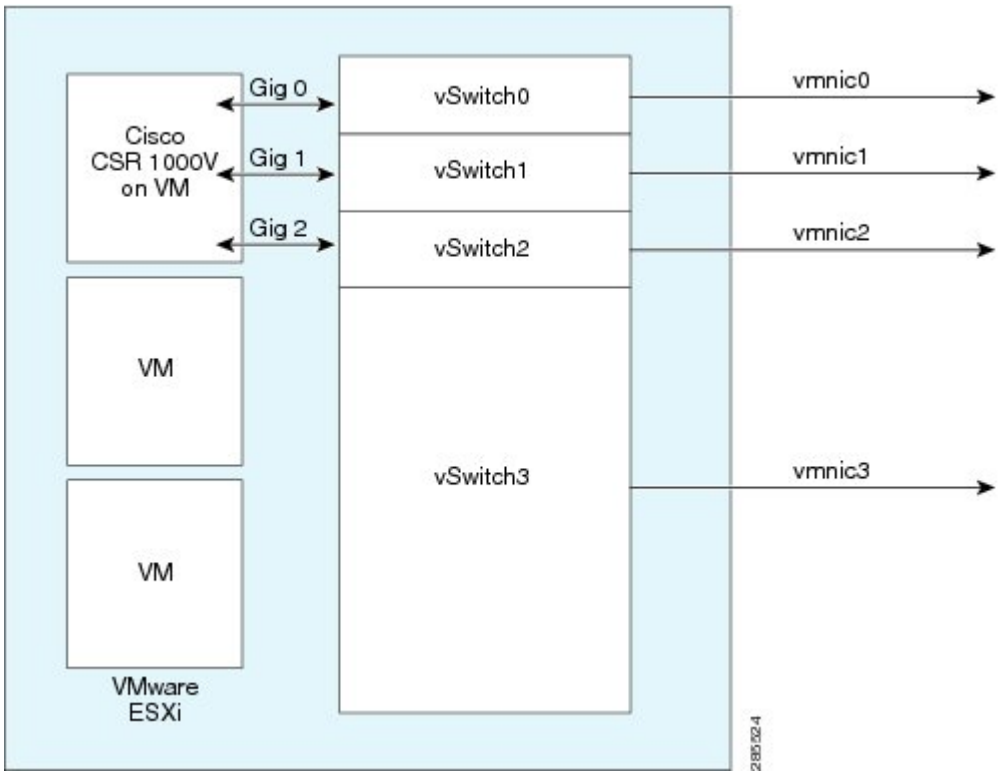
Step 5 Configure the router interfaces on the cloned VM accordingly.

If you follow these steps, the router configuration on the cloned VM should match the configuration of the original VM.

Mapping Cisco CSR 1000v Network Interfaces with vSwitch Interfaces

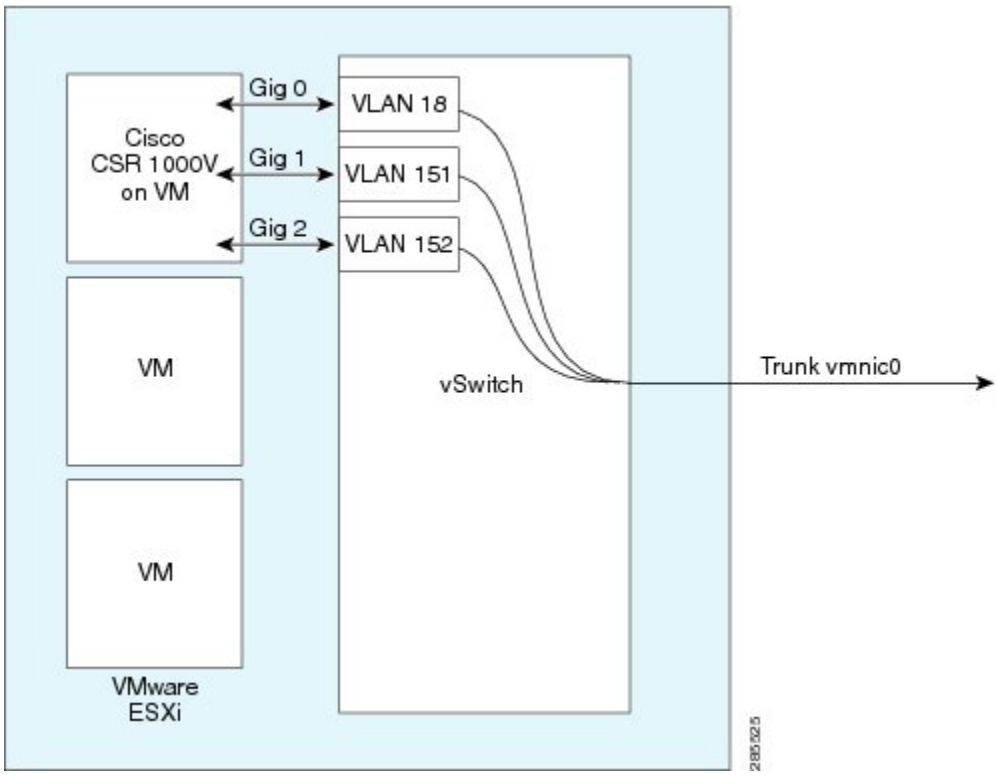
You can configure the network interfaces in ESXi in different ways to accommodate the Cisco CSR 1000v interfaces. The figure below shows an example where each Cisco CSR 1000v router interface is mapped to one host Ethernet interface.

Figure 13: Cisco CSR 1000v Interfaces Mapped to Individual ESXi Host Ethernet Interfaces



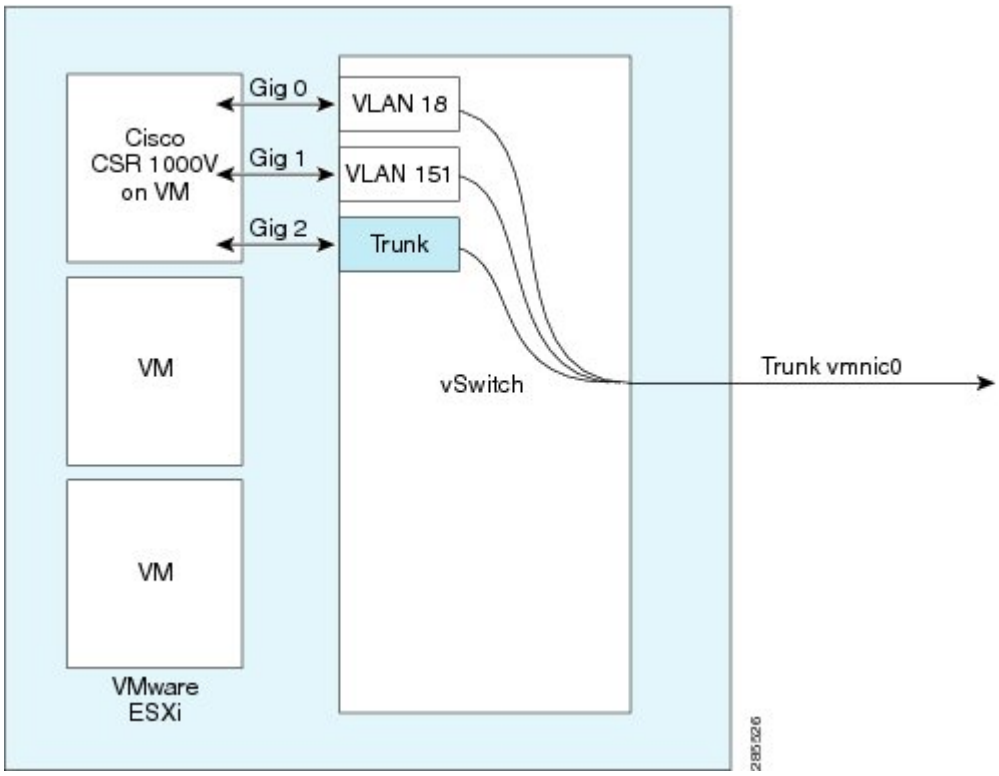
The figure below shows an example with multiple Cisco CSR 1000v interfaces sharing one host ESXi Ethernet interface.

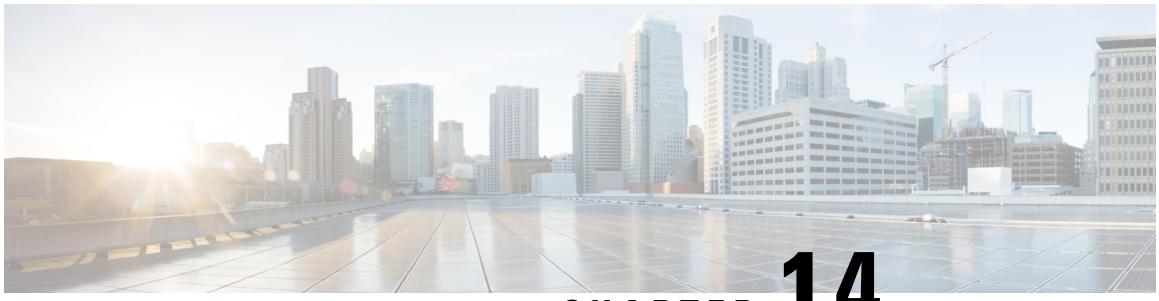
Figure 14: Cisco CSR 1000v Interfaces Sharing One ESXi Host Ethernet Interface



The next figure shows one of the Cisco CSR 1000v interfaces mapped directly to a trunk interface on the vSwitch.

Figure 15: Cisco CSR 1000v Interfaces Directly Mapped to vSwitch Trunk





CHAPTER 14

Configuring the vCPU Distribution across the Data, Control and Service Planes

Information About vCPU Allocation and Distribution

You can allocate and distribute the vCPUs of the following planes: Control Plane(CP), Data Plane(DP), and Service Plane(SP) by using templates. Note that the Service Plane includes containers running SNORT.

Use one of the following templates for vCPU distribution:

vCPU Distribution: Control Plane Extra heavy

The following table shows the vCPU distribution for the Control Plane Extra heavy template.

Table 36: Control Plane Extra heavy—vCPU Distribution

Number of vCPUs	1	2	4	8
Control Plane	1/3	1/2	1 1/2	1 1/2
Service Plane	1/3	1/2	1 1/2	1 1/2
Data Plane	1/3	1	1	5

Note Using an Control Plane Extra heavy template, a service plane app can obtain 1.5 full cores for its operation. Example: WAAS.

vCPU Distribution: Control Plane heavy

The following table shows the vCPU distribution for the Control Plane heavy template.

Table 37: Control Plane heavy—vCPU Distribution

Number of vCPUs	1	2	4	8
Control Plane	1/3	1/2	1	1
Service Plane	1/3	1/2	1	1
Data Plane	1/3	1	2	6

Note The Control Plane heavy template allocates an extra core to the Control Plane/Service Plane services, compared to the Data Plane heavy template (there is one core for the Control Plane and another core for the Service Plane). If there is no Service Plane application, the Control Plane can utilize all of the resources (2 cores).

vCPU Distribution: Data Plane heavy

Note The Data Plane heavy template is the default vCPU Distribution template. Even if the configuration output reads 'None' as the value for the Template option, the Data Plane heavy template is still applied.

The above mentioned statement is not applicable for Cisco CSR1000V and Cisco ISRv instances running in the controller mode.

The following table shows the vCPU distribution for the Data Plane heavy template.

Table 38: Data Plane heavy—vCPU Distribution

Number of vCPUs	1	2	4	8
Control Plane	1/3	1/2	1/2	1/2
Service Plane	1/3	1/2	1/2	1/2
Data Plane	1/3	1	3	7

Note By default, the Cisco CSR 1000v core allocation favors a larger data plane for performance. If there is no Service Plane application, the Control Plane can utilize the Service Plane's resources.

vCPU Distribution: Data Plane normal

You can use the vCPU distribution for the Data Plane normal template to force the Cisco CSR 1000v to behave in the same way as before using a template for vCPU distribution.

For example, after creating a Cisco CSR 1000v VM using the Data Plane heavy template for vCPU distribution, specified in the ovf-env.xml file, you can later use CLI commands in the Data Plane normal template to override the XML file settings that were previously applied by the Data Plane heavy template.

vCPU Distribution: Service Plane heavy

The following table shows the vCPU distribution for the Service Plane heavy template.

Table 39: Service Plane heavy - vCPU Distribution

Number of vCPUs	1	2	4	8
Control Plane	1/3	1/2	1	2
Service Plane	1/3	1/2	1	2
Data Plane	1/3	1	2	4

Note Using a Service Plane heavy template, a Service Plane application (such as Snort IPS) can use up to 2 full cores for its operation.

vCPU Distribution: Service Plane medium

The following table shows the vCPU distribution for the Service Plane medium template.

Table 40: Service Plane medium—vCPU Distribution

Number of vCPUs	1	2	4	8
Control Plane	1/3	1/2	1	1
Service Plane	1/3	1/2	1	1
Data Plane	1/3	1	2	6

How to Boot the Cisco CSR 1000v with an OVA image

To boot the Cisco CSR 1000v with an OVA image, boot from a CDROM containing the ovf-env.xml file. This XML file contains the vCPU distribution templates. A template is simply an additional bootstrap property. For more information about bootstrap properties, see Bootstrapping the CSR Configuration.

This is an example of the part of the XML file that specifies a Service Plane medium template: `<Property oe:key="com.cisco.csr1000v.resource-template.1" oe:value="service_plane_medium"/>`

How to Configure vCPU Distribution across the Data, Control and Service Planes

Enter the `platform resource` command on the Cisco CSR 1000v to select a template for vCPU distribution.

configure template

platform resource *template*

Example:

```
Router# configure template
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# platform resource ?
  control-plane-extra-heavy Use Control Plane Extra Heavy template
  control-plane-heavy   Use Control Plane Heavy template
  data-plane-heavy      Use Data Plane Heavy template
  data-plane-normal     Use Data Plane Normal template
  service-plane-heavy   Use Service Plane Heavy template
  service-plane-medium  Use Service Plane Medium template
Router(config)# platform resource service-plane-heavy
```

Note After entering the `platform resource` command, you must reboot the Cisco CSR 1000v to activate the template.

Determine the Active vCPU Distribution Template

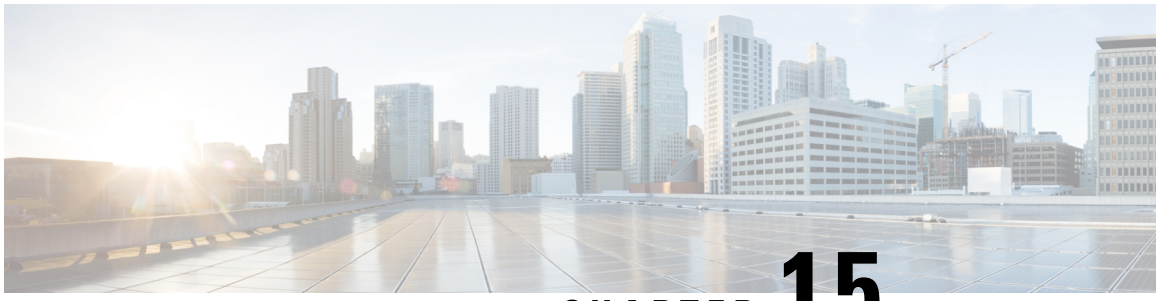
To determine which template is being used for vCPU distribution, use the following command:

show platform software cpu alloc

Example:

```
Router# show platform software cpu alloc
CPU alloc information:
Control plane cpu alloc: 0-1
Data plane cpu alloc: 2-3
Service plane cpu alloc: 0-1
Template used: CLI-service_plane_heavy
```

Note The Control plane and the Service plane share cores 0 and 1.



CHAPTER 15

Accessing and Using GRUB Mode

About GRUB Mode and the Configuration Register

The Cisco CSR 1000V/ISRv has a 16-bit configuration register in NVRAM. Each bit has value 1 (on or set) or value 0 (off or clear), and each bit setting affects the router behavior upon the next reload power cycle. The GRUB mode supports a subset of configuration register options compared to ROMMON options on other Cisco routers.

You can use the configuration register to:

- Force the router to boot into the GRUB (bootstrap program)
- Select a boot source and default boot filename
- Recover a lost password

The table below describes the configuration register bits.

Table 41: Configuration Register Bit Descriptions

BitNumber	Hexadecimal	Meaning
00–03	0x0000–0x000F	Boot field. The boot field setting determines whether the router loads an operating system and where it obtains the system image. See the table "Boot Field Configuration Register Bit Descriptions" for details.

BitNumber	Hexadecimal	Meaning
06	0x0040	Causes the system software to ignore the contents of NVRAM. This can be used for password recovery.

Note Entering the GRUB mode for Cisco CSR1000V running on cloud solutions depends on the console access capabilities of the cloud provider. If the cloud provider provides limited access to console, you cannot access the GRUB mode for password recovery.

The next table describes the boot field, which is the lowest four bits of the configuration register (bits 3, 2, 1, and 0). The boot field setting determines whether the router loads an operating system.

Table 42: Boot Field Configuration Register Bit Descriptions

Boot Field(Bits 3, 2, 1, and 0)	Meaning
0000 (0x0)	At the next power cycle or reload, the router boots to the GRUB (bootstrap program). In GRUB mode, you must manually boot the system image or any other image by using the **boot** command.
0001 - 1111 (0x01 - 0x0F)	At the next power cycle or reload, the router sequentially processes each **boot system** command in global configuration mode that is stored in the configuration file until the system boots successfully. If the **no boot** system commands are stored in the configuration file, or if these commands are not executed successfully, GRUB dictates the image that needs to be booted.

Note Use the 0x000 setting to configure the router to automatically enter GRUB mode when the router reboots.

Accessing GRUB Mode

Perform the following step to access GRUB mode:

SUMMARY STEPS

1. **enable**
2. **config-register 0x0000**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**config-register 0x0000** **Example:** `Router# config-register 0x0000`	Enters the GRUB mode by entering the "0000" value (0x0).

What to do next

The following shows an example of entering GRUB mode.

```
Router(config)# config-register 0x0000

GNU GRUB  version 0.97  (638K lower / 3143616K upper memory)
 [ Minimal BASH-like line editing is supported.  For the first word, TAB
   lists possible command completions.  Anywhere else TAB lists the possible
   completions of a device/filename.  ESC at any time exits to menu. ]
grub> help
 [ Minimal BASH-like line editing is supported.  For the first word, TAB
   lists possible command completions.  Anywhere else TAB lists the possible
   completions of a device/filename.  ESC at any time exits to menu. ]
confreg [VALUE]                       help [--all] [PATTERN ...]
grub>
```

If you enter a question mark at the grub> prompt, the system shows you the two options available, for either viewing the system help or for entering the **confreg** command.

Using the GRUB Menu

The GRUB menu is used to display the software images loaded on the router, and to select which image to boot from. To access the GRUB menu, enter **ESC** at the GRUB prompt. The following shows the GRUB menu display.

```
GNU GRUB  version 0.97  (638K lower / 3143616K upper memory)
 +-------------------------------------------------------------------------+
 | CSR1000v - csr1000v-universalk9.03.10.00.S.153-3.S-ext.SPA.bin          |
 | CSR1000v - packages.conf                                                |
 | CSR1000v - GOLDEN IMAGE                                                 |
 |                                                                         |
 |                                                                         |
 |                                                                         |
 |                                                                         |
 |                                                                         |
 |                                                                         |
```

```
 |                                                                          |
 |                                                                          |
 |                                                                          |
 +--------------------------------------------------------------------------+
      Use the ^ and v keys to select which entry is highlighted.
      Press enter to boot the selected OS, or 'c' for a command-line.
```

Select the image to boot the router from using the up and down arrow key. To return to the GRUB prompt, enter the letter **c**.

Modifying the Configuration Register (confreg)

This section describes how to modify the configuration register by using the **confreg** GRUB command. This command is similar to the **confreg** ROMMON command on other Cisco hardware routers. Because the router does not include a ROMMON mode, the similar functionality is handled in GRUB command mode.

You can also modify the configuration register setting from the Cisco IOS XE CLI by using the **config-register** command in global configuration mode.

Note The modified configuration register value is automatically written into NVRAM, but the new value does not take effect until you reset or power-cycle the router.

SUMMARY STEPS

1. **confreg** [*value*]

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**confreg** [*value*] **Example:** `grub> confreg 0x2102`	Changes the configuration register settings while in GRUB command mode. • Optionally, enter the new hexadecimal value for the configuration register. The value range is from 0x0 to 0xFFFF. • If you do not enter the value, the router prompts for each bit of the 16-bit configuration register.

What to do next

The following shows an example of entering GRUB mode and using the configuration register. You access the GRUB mode by entering the Cisco IOS XE **config-register** command and specifying the value as “0000”.

```
Router(config)# config-register 0x0000

GNU GRUB  version 0.97  (638K lower / 3143616K upper memory)
```

```
 [ Minimal BASH-like line editing is supported.  For the first word, TAB
   lists possible command completions.  Anywhere else TAB lists the possible
   completions of a device/filename.  ESC at any time exits to menu. ]
grub> help
 [ Minimal BASH-like line editing is supported.  For the first word, TAB
   lists possible command completions.  Anywhere else TAB lists the possible
   completions of a device/filename.  ESC at any time exits to menu. ]
confreg [VALUE]                       help [--all] [PATTERN ...]
grub> confreg
           Configuration Summary
   (Virtual Configuration Register: 0x0)
enabled are:
boot: the boot loader
do you wish to change the configuration? y/n [n
]:
ignore system config info? y/n [n
]:
automatically boot default system image? y/n [n
]:
Configuration Register: 0x0
grub> confreg
           Configuration Summary
   (Virtual Configuration Register: 0x0)
enabled are:
boot: the boot loader
do you wish to change the configuration? y/n [n
]:
ignore system config info? y/n [n]:
automatically boot default system image? y/n [n]:
Configuration Register: 0x42
grub> confreg 0x2102
Configuration Register: 0x2102
grub> confreg
           Configuration Summary
   (Virtual Configuration Register: 0x2102)
enabled are:
boot: default image
do you wish to change the configuration? y/n [n
]:
grub>
grub>
    GNU GRUB  version 0.97  (638K lower / 3143616K upper memory)
------------------------------------------------------------------------
 0: CSR1000v - packages.conf
 1: CSR1000v - csr100v-packages-universalk9
 2: CSR1000v - GOLDEN IMAGE
------------------------------------------------------------------------
      Use the ^ and v keys to select which entry is highlighted.
      Press enter to boot the selected OS, or 'c' for a command-line.
    Highlighted entry is 0:
  Booting 'CSR1000v - packages.conf'
root (hd0,0)
 Filesystem type is ext2fs, partition type 0x83
kernel /packages.conf rw root=/dev/ram console=ttyS1,9600 max_loop=64 HARDWARE=
virtual SR_BOOT=harddisk:packages.conf
Calculating SHA-1 hash...done
SHA-1 hash:
        calculated   817e1716:e8e62778:7dd0b806:32db2bdd:13e51407
        expected     817e1716:e8e62778:7dd0b806:32db2bdd:13e51407
package header rev 1 structure detected
Calculating SHA-1 hash...done
SHA-1 hash:
        calculated   d4eaba99:34cbda63:26151233:9d0e9aa4:9c625302
        expected     d4eaba99:34cbda63:26151233:9d0e9aa4:9c625302
```

```
Package type:0x7531, flags:0x0
   [Linux-bzImage, setup=0x2e00, size=0x2c18c00]
   [isord @ 0x7e6d0000, 0x191f000 bytes]
```

Changing the Configuration Register Settings

You can change the configuration register settings from either the GRUB or the Cisco IOS XE CLI. This section describes how to modify the configuration register settings from the Cisco IOS XE CLI.

To change the configuration register settings from the Cisco IOS XE CLI, complete the following steps:

SUMMARY STEPS

1. Power on the router.
2. If you are asked whether you would like to enter the initial dialog, answer no:
3. Enter privileged EXEC mode by typing enable and, if prompted, enter your password:
4. Enter global configuration mode:
5. To change the configuration register settings, enter the **config-register** *value* command, where *value* is a hexadecimal number preceded by **0x**:
6. Exit global configuration mode:
7. Save the configuration changes to NVRAM:

DETAILED STEPS

Procedure

Step 1 Power on the router.

Step 2 If you are asked whether you would like to enter the initial dialog, answer no:

Example:

```
Would you like to enter the initial dialog? [yes]: no
```

After a few seconds, the user EXEC prompt (Router>) appears.

Step 3 Enter privileged EXEC mode by typing enable and, if prompted, enter your password:

Example:

```
Router> enable
Password: password
Router#
```

Step 4 Enter global configuration mode:

Example:

```
Router# configure terminal
Enter configuration commands, one per line.
Edit with DELETE, CTRL/W, and CTRL/U; end with CTRL/Z
```

Step 5 To change the configuration register settings, enter the **config-register** *value* command, where *value* is a hexadecimal number preceded by **0x**:

Example:

```
Router(config)# config-register 0x
value
```

Step 6 Exit global configuration mode:

Example:

```
Router(config)# end
Router#
```

Step 7 Save the configuration changes to NVRAM:

Router# **copy running-config startup-config**

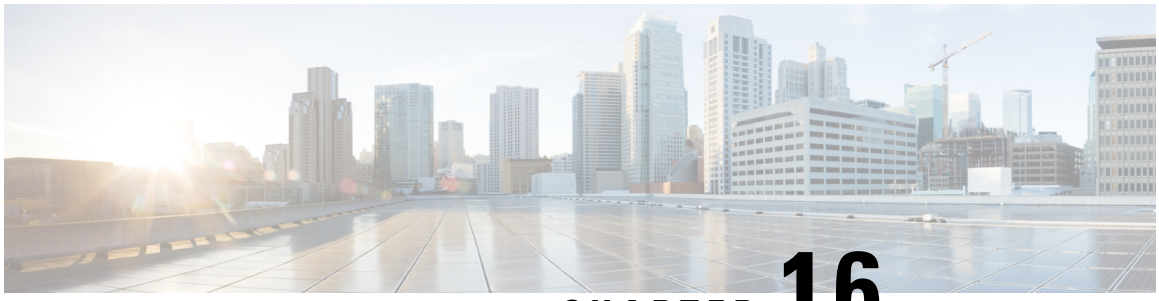
The new configuration register settings are saved to NVRAM, but they do not take effect until the next router reload or power cycle.

Displaying the Configuration Register Settings

To display the configuration register settings that are currently in effect and the settings that will be used at the next router reload, enter the **show version** command in privileged EXEC mode.

The configuration register settings are displayed in the last line of the **show version** command output:

```
Configuration register is 0x142 (will be 0x142 at next reload)
```



CHAPTER 16

Configuring Call Home for the Cisco CSR 1000v

Prerequisites for Call Home

The Call Home feature provides email-based and web-based notification of critical system events. A versatile range of message formats are available for optimal compatibility with pager services, standard email, or XML-based automated parsing applications. Common uses of this feature may include direct paging of a network support engineer, email notification to a network operations center, XML delivery to a support website, and use of Cisco Smart Call Home services for direct case generation with the Cisco Systems Technical Assistance Center (TAC).

Note The router supports the Call Home feature beginning with Cisco IOS XE Release 3.12S.

Consider the following points before you configure Call Home:

- Contact email address (required for full registration with Smart Call Home, optional if Call Home is enabled in anonymous mode), phone number (optional), and street address information (optional) should be configured so that the receiver can determine the origin of messages received.
- At least one destination profile (predefined or user-defined) must be configured. The destination profile you use depends on whether the receiving entity is a pager, an email address, or an automated service such as Cisco Smart Call Home.

- If the destination profile uses email message delivery, you must specify a Simple Mail Transfer Protocol (SMTP) server. Configuring the trustpoint CA is not required for HTTPS server connection since the trustpool feature enabled by default.
- The router must have IP connectivity to an email server or the destination HTTP(S) server.
- If Cisco Smart Call Home is used, an active service contract covering the device is required to provide full SCH service.

Information About Call Home

The Call Home feature can deliver alert messages containing information on configuration, inventory, syslog, snapshot, and crash events. It provides these alert messages as either email-based or web-based messages. Multiple message formats are available, allowing for compatibility with pager services, standard email, or XML-based automated parsing applications. This feature can deliver alerts to multiple recipients, which are Call Home destination profiles. Each destination profile has configurable message formats and content categories. A predefined destination profile is provided for sending alerts to the Cisco Smart Call Home server. The predefined profile defines both the email address and the HTTP(S) URL; the transport method configured in the profile determines whether the email address or the HTTP(S) URL is used.

Flexible message delivery and format options make it easy to integrate specific support requirements.

Benefits of Using Call Home

The Call Home feature offers the following benefits:

- Multiple message-format options
  - Short Text—Suitable for pagers or printed reports.
  - Long Text—Full formatted message information suitable for human reading.
  - XML—Machine-readable format using XML. The XML format enables communication with the Cisco TAC.
- Multiple concurrent message destinations
- Multiple message categories including configuration, inventory, syslog, snapshot, and crash events
- Filtering of messages by severity and pattern matching
- Scheduling of periodic message sending

Obtaining Smart Call Home Services

If you have a service contract directly with Cisco, you can register for the Smart Call Home service. Smart Call Home analyzes Call Home messages and provides background information and recommendations. For critical issues, Automatic Service Requests are generated with the Cisco TAC.

Smart Call Home offers the following features:

- Continuous device health monitoring and real-time alerts.
- Analysis of Smart Call Home messages and, if needed, Automatic Service Request generation routed to the correct TAC team, including detailed diagnostic information to speed problem resolution.

- Secure message transport directly from your device or through an HTTP proxy server or a downloadable Transport Gateway (TG). You can use a TG aggregation point to support multiple devices or in cases where security dictates that your devices may not be connected directly to the Internet.
- Web-based access to Smart Call Home messages and recommendations, inventory, and configuration information for all Smart Call Home devices provides access to associated field notices, security advisories, and end-of-life information.

You need the following items to register for Smart Call Home:

- SMARTnet contract number for your router
- Your email address
- Your Cisco.com username

For detailed information on Smart Call Home, see www.cisco.com/go/smartcallhome/index.html.

Anonymous Reporting

Smart Call Home is a service capability included with many Cisco service contracts and helps you to resolve problems. In addition, information is gained from; for example, crash messages to help Cisco understand issues occurring in the field. If you decide not to use Smart Call Home, you can still enable Anonymous Reporting to allow Cisco to securely receive minimal error and health information from the device. If you enable Anonymous Reporting, your customer identity will remain anonymous, and no identifying information is sent.

Note When you enable Anonymous Reporting, you acknowledge your consent to transfer the specified data to Cisco or to vendors operating on behalf of Cisco (including countries outside the United States). Cisco maintains the privacy of all customers. For information about how Cisco treats personal information, see the Cisco Online Privacy Statement .

When Call Home is configured in an anonymous way, only crash, inventory, and test messages are sent to Cisco. No identifying information is sent.

For more information about what is sent in these messages, see the Alert Group Trigger Events and Commands, on page 309.

How to Configure Call Home

How to Configure Call Home

The following sections show how you can configure Call Home using a single command:

- Configuring Smart Call Home (Single Command), on page 274
- Configuring and Enabling Smart Call Home, on page 275

The following sections show detailed or optional configurations:

- Enabling and Disabling Call Home , on page 275

Configuring Smart Call Home (Single Command)

To enable all Call Home basic configurations using a single command, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home reporting** {**anonymous** | **contact-email-addr** *email-address*} [**http-proxy** {*ipv4-address* | *ipv6-address* | *name*} **port** *port-number*]

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**call-home reporting** {**anonymous** | **contact-email-addr** *email-address*} [**http-proxy** {*ipv4-address* | *ipv6-address* | *name*} **port** *port-number*] **Example:** `Router(config)# call-home reporting contact-email-addr email@company.com`	Enables all Call Home basic configurations using a single command. • **anonymous**—Enables Call-Home TAC profile to only send crash, inventory, and test messages and send the messages in an anonymous way. • **contact-email-addr**—Enables Smart Call Home service full reporting capability and sends a full inventory message from Call-Home TAC profile to Smart Call Home server to start full registration process.

	Command or Action	Purpose
		• **http-proxy** {*ipv4-address* | *ipv6-address* | *name*}—An ipv4 or ipv6 address or server name. Maximum length is 64. • **port** *port-number*—Port number. Range is 1 to 65535. **Note** HTTP proxy option allows you to make use of your own proxy server to buffer and secure internet connections from your devices. **Note** After successfully enabling Call Home either in anonymous or full registration mode using the **call-home reporting** command, an inventory message is sent out. If Call Home is enabled in anonymous mode, an anonymous inventory message is sent out. If Call Home is enabled in full registration mode, a Full Inventory message for full registration mode is sent. For more information about what is sent in these messages, see the Alert Group Trigger Events and Commands, on page 309.

Configuring and Enabling Smart Call Home

For application and configuration information about the Cisco Smart Call Home service, see the Smart Call Home User Guide . See also the Cisco Support Community page for Smart Call Home.

The user guide includes configuration examples for sending Smart Call Home messages directly from your device or through a transport gateway (TG) aggregation point.

Note For security reasons, we recommend that you use the HTTPS transport options, due to the additional payload encryption that HTTPS offers. The Transport Gateway software is downloadable from Cisco.com and is available if you require an aggregation point or a proxy for connection to the Internet.

The implementation on the Cisco CSR 1000v/ISRv supports the trustpool feature (embedded CA certificates in IOS images). The trustpool feature simplifies configuration to enable Smart Call Home service on configured devices. It eliminates the requirement of manually configuring the trustpoint and provides automatic update of the CA certificate should it change in the future.

Enabling and Disabling Call Home

SUMMARY STEPS

1. **configure terminal**
2. **service call-home**
3. **no service call-home**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**service call-home** **Example:** `Router(config)# service call-home`	Enables the Call Home feature. By default, Call Home is disabled.
Step 3	**no service call-home** **Example:** `Router(config)# no service call-home`	Disables the Call Home feature.

Configuring Contact Information

Each router must include a contact email address (except if Call Home is enabled in anonymous mode). You can optionally include a phone number, street address, contract ID, customer ID, and site ID.

To assign the contact information, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. **contact-email-addr** *email-address*
4. **phone-number** +*phone-number*
5. **street-address** *street-address*
6. **customer-id** *text*
7. **site-id** *text*
8. **contract-id** *text*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	Purpose
	`Router# configure terminal`	
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters the Call Home configuration submode.
Step 3	**contact-email-addr** *email-address* **Example:** `Router(cfg-call-home)# contact-email-addr username@example.com`	Designates your email address. Enter up to 200 characters in email address format with no spaces.
Step 4	**phone-number** +*phone-number* **Example:** `Router(cfg-call-home)# phone-number +1-800-555-4567`	(Optional) Assigns your phone number. **Note** The number must begin with a plus (**+**) prefix and may contain only dashes (-) and numbers. Enter up to 17 characters. If you include spaces, you must enclose your entry in quotes (“”).
Step 5	**street-address** *street-address* **Example:** `Router(cfg-call-home)# street-address “1234 Picaboo Street, Any city, Any state, 12345“`	(Optional) Assigns your street address where RMA equipment can be shipped. Enter up to 200 characters. If you include spaces, you must enclose your entry in quotes (“”).
Step 6	**customer-id** *text* **Example:** `Router(cfg-call-home)# customer-id Customer1234`	(Optional) Identifies customer ID. Enter up to 64 characters. If you include spaces, you must enclose your entry in quotes (“”).
Step 7	**site-id** *text* **Example:** `Router(cfg-call-home)# site-id Site1ManhattanNY`	(Optional) Identifies customer site ID. Enter up to 200 characters. If you include spaces, you must enclose your entry in quotes (“”).
Step 8	**contract-id** *text* **Example:** `Router(cfg-call-home)# contract-id Company1234`	(Optional) Identifies your contract ID for the router. Enter up to 64 characters. If you include spaces, you must enclose your entry in quotes (“”).

Information About Destination Profiles

A destination profile contains the required delivery information for an alert notification. At least one destination profile is required. You can configure multiple destination profiles of one or more types.

You can create and define a new destination profile or copy and use the predefined destination profile. If you define a new destination profile, you must assign a profile name.

You can control which profile to be used for Smart Licensing by enabling or disabling smart-licensing data of that profile. Only one active profile can have smart-license data enabled. For more information about Smart Licensing, see Installing Cisco CSR 1000v Licenses, on page 161.

Note If you use the Cisco Smart Call Home service, the destination profile must use the XML message format.

You can configure the following attributes for a destination profile:

- Profile name—String that uniquely identifies each user-defined destination profile. The profile name is limited to 31 characters and is not case-sensitive. You cannot use **all** as a profile name.
- Transport method—Transport mechanism, either email or HTTP (including HTTPS), for delivery of alerts.

  For both the CiscoTAC-1 profile and user-defined destination profiles, email is the default, and you can enable either or both transport mechanisms. If you disable both methods, email is enabled.

  For the predefined CiscoTAC-1 profile, you can enable either transport mechanism, but not both.
- Destination address—The actual address related to the transport method by which the alert should be sent.

In this version of the Call Home feature, you can change the destination of the CiscoTAC-1 profile.

- Message formatting—The message format used for sending the alert. The format options for a user-defined destination profile are long-text, short-text, or XML. The default is XML. For the predefined CiscoTAC-1 profile, only XML is allowed.
- Message size—The maximum destination message size. The valid range is 50 to 3,145,728 bytes. The default is 3,145,728 bytes.
- Reporting method—You can choose which data to report for a profile. You can enable reporting of Smart Call Home data or Smart Licensing Data, or both. Only one active profile is allowed to report Smart Licensing data at a time.
- Anonymous reporting—You can choose for your customer identity to remain anonymous, and no identifying information is sent.
- Subscribing to interesting alert-groups—You can choose to subscribe to alert-groups highlighting your interests.

This section contains the following subsections:

Creating a New Destination Profile

To create and configure a new destination profile, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**

3. **profile** *name*
4. [**no**] **destination transport-method** {**email** | **http**}
5. **destination address** {**email** *email-address* | **http** *url*}
6. **destination preferred-msg-format** {**long-text** | **short-text** | **xml**}
7. **destination message-size-limit** *bytes*
8. **active**
9. **reporting** {**all** | **smart-call-home-data**| **smart-licensing-data** }
10. **end**
11. **show call-home profile** {*name* | **all**}
12. **show call-home smart-licensing**
13. **show call-home smart-licensing statistics**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters the Call Home configuration submode.
Step 3	**profile** *name* **Example:** `Router(config-call-home)# profile profile1`	Enters the Call Home destination profile configuration submode for the specified destination profile. If the specified destination profile does not exist, it is created.
Step 4	[**no**] **destination transport-method** {**email**	**http**} **Example:** `Router(cfg-call-home-profile)# destination transport-method email`	(Optional) Enables the message transport method. The **no** option disables the method.
Step 5	**destination address** {**email** *email-address*	**http** *url*} **Example:** `Router(cfg-call-home-profile)# destination address email myaddress@example.com`	Configures the destination email address or URL to which Call Home messages are sent. **Note** When entering a destination URL, include either **http://** or **https://**, depending on whether the server is a secure server.

	Command or Action	Purpose
Step 6	**destination preferred-msg-format** {**long-text** | **short-text** | **xml**} **Example:** `Router(cfg-call-home-profile)# destination preferred-msg-format xml`	(Optional) Configures a preferred message format. The default is XML.
Step 7	**destination message-size-limit** *bytes* **Example:** `Router(cfg-call-home-profile)# destination message-size-limit 3145728`	(Optional) Configures a maximum destination message size for the destination profile.
Step 8	**active** **Example:** `Router(cfg-call-home-profile)# active`	Enables the destination profile. By default, the profile is enabled when it is created.
Step 9	**reporting** {**all** | **smart-call-home-data**| **smart-licensing-data** }	Configures the type of data to report for a profile. You can select either to report Smart Call Home data or Smart Licensing data. Selecting the all option reports data for both types of data.
Step 10	**end** **Example:** `Router(cfg-call-home-profile)# end`	Returns to privileged EXEC mode.
Step 11	**show call-home profile** {*name* | **all**} **Example:** `Router# show call-home profile profile1`	Displays the destination profile configuration for the specified profile or all configured profiles.
Step 12	**show call-home smart-licensing**	Displays the current Call Home Smart Licensing settings for the configured destination profiles.
Step 13	**show call-home smart-licensing statistics**	Displays the Call Home Smart Licensing statistics.

Copying a Destination Profile

To create a new destination profile by copying an existing profile, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. **copy profile** *source-profile target-profile*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters the Call Home configuration submode.
Step 3	**copy profile** *source-profile target-profile* **Example:** `Router(cfg-call-home)# copy profile profile1 profile2`	Creates a new destination profile with the same configuration settings as the existing destination profile.

Setting Profiles to Anonymous Mode

To set an anonymous profile, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. **profile** *name*
4. **anonymous-reporting-only**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters Call Home configuration submode.

	Command or Action	Purpose
Step 3	**profile** *name* **Example:** `Router(cfg-call-home) profile CiscoTAC-1`	Enables profile configuration mode.
Step 4	**anonymous-reporting-only** **Example:** `Router(cfg-call-home-profile)#` **`anonymous-reporting-only`**	Sets the profile to anonymous mode. **Note** By default, the profile sends a full report of all types of events subscribed in the profile. When **anonymous-reporting-only** is set, only crash, inventory, and test messages are sent.

Subscribing to Alert Groups

An alert group is a predefined subset of Call Home alerts supported in all routers. Different types of Call Home alerts are grouped into different alert groups depending on their type. The following alert groups are available:

- Configuration
- Inventory
- Syslog
- Crash
- Snapshot

This section contains the following subsections:

- Periodic Notification, on page 285
- Message Severity Threshold, on page 285
- Configuring Snapshot Command List, on page 286

The triggering events for each alert group are listed in Alert Group Trigger Events and Commands, on page 309, and the contents of the alert group messages are listed in Message Contents, on page 310.

You can select one or more alert groups to be received by a destination profile.

Note A Call Home alert is sent only to destination profiles that have subscribed to the alert group containing that Call Home alert. In addition, the alert group must be enabled.

To subscribe a destination profile to one or more alert groups, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**

3. **alert-group** {**all** | **configuration** | **environment** | **inventory** | **syslog** | crash | snapshot}
4. **profile** *name*
5. **subscribe-to-alert-group configuration**[**periodic** {**daily** *hh:mm* | **monthly** *date hh:mm* | **weekly** *day hh:mm*}]
6. **subscribe-to-alert-group inventory** *[***periodic** *{***daily** *hh:mm* | **monthly** *date hh:mm* | **weekly** *day hh:mm}]*
7. **subscribe-to-alert-group syslog**[**severity** {**catastrophic** | **disaster** |**fatal** | **critical** | **major** | **minor** | **warning** |**notification** | **normal** | **debugging**}][**pattern** *string*]
8. **subscribe-to-alert-group crash**
9. **subscribe-to-alert-group snapshot** [**periodic** {**daily** *hh:mm* | **hourly** *mm* | **interval** *mm* | **monthly** *date hh:mm* | **weekly** *day hh:mm*}]
10. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters Call Home configuration submode.
Step 3	**alert-group** {**all**	**configuration**	**environment**	**inventory**	**syslog**	crash	snapshot} **Example:** `Router(cfg-call-home)# alert-group all`	Enables the specified alert group. Use the keyword all to enable all alert groups. By default, all alert groups are enabled.
Step 4	**profile** *name* **Example:** `Router(cfg-call-home)# profile profile1`	Enters Call Home destination profile configuration submode for the specified destination profile.
Step 5	**subscribe-to-alert-group configuration**[**periodic** {**daily** *hh:mm*	**monthly** *date hh:mm*	**weekly** *day hh:mm*}] **Example:** `Router(cfg-call-home-profile)# subscribe-to-alert-group configurationperiodic daily 12:00`	Subscribes this destination profile to the Configuration alert group. The Configuration alert group can be configured for periodic notification, as described in Periodic Notification, on page 285.

	Command or Action	Purpose
Step 6	**subscribe-to-alert-group inventory** *[***periodic** *{***daily** *hh:mm* | **monthly** *date hh:mm* | **weekly** *day hh:mm}]* **Example:** `Router(cfg-call-home-profile)# subscribe-to-alert-group inventory periodic monthly 1 12:00`	Subscribes this destination profile to the Inventory alert group. The Inventory alert group can be configured for periodic notification, as described in Periodic Notification, on page 285.
Step 7	**subscribe-to-alert-group syslog**[**severity** {**catastrophic** | **disaster** |**fatal** | **critical** | **major** | **minor** | **warning** |**notification** | **normal** | **debugging**}][**pattern** *string*] **Example:** `Router(cfg-call-home-profile)# subscribe-to-alert-group syslog severity major`	Subscribes this destination profile to the Syslog alert group. The Syslog alert group can be configured to filter messages based on severity, as described in Message Severity Threshold, on page 285. You can specify a text pattern to be matched within each syslog message. If you configure a pattern, a Syslog alert group message is sent only if it contains the specified pattern and meets the severity threshold. If the pattern contains spaces, you must enclose it in quotes (“”). You can specify up to five patterns for each destination profile.
Step 8	**subscribe-to-alert-group crash** **Example:** `Router(cfg-call-home-profile)# subscribe-to-alert-group crash`	Subscribes to the Crash alert group in user profile. By default, the CiscoTAC-1 profile subscribes to the Crash alert group and cannot be unsubscribed.
Step 9	**subscribe-to-alert-group snapshot** [**periodic** {**daily** *hh:mm* | **hourly** *mm* | **interval** *mm* | **monthly** *date hh:mm* | **weekly** *day hh:mm*}] **Example:** `Router(cfg-call-home-profile)# subscribe-to-alert-group snapshot periodic daily 12:00`	Subscribes this destination profile to the Snapshot alert group. The Snapshot alert group can be configured for periodic notification, as described in Periodic Notification, on page 285. By default, the Snapshot alert group has no command to run. You can add commands into the alert group, as described in Configuring Snapshot Command List, on page 286. In doing so, the output of the commands added in the Snapshot alert group will be included in the snapshot message.
Step 10	**end** **Example:** `Router(cfg-call-home-profile)# end`	Exits configuration mode.

What to do next

Note As an alternative to subscribing to individual alert groups, you can subscribe to all alert groups by entering the **subscribe-to-alert-group all** command. However, entering this command causes a large number of syslog messages to generate. We recommend subscribing to alert groups individually, using appropriate severity levels and patterns when possible.

Periodic Notification

When you subscribe a destination profile to the Configuration, Inventory, or Snapshot alert group, you can choose to receive the alert group messages asynchronously or periodically at a specified time. The sending period can be one of the following:

- Daily—Specifies the time of day to send, using an hour:minute format hh:mm, with a 24-hour clock (for example, 14:30).
- Weekly—Specifies the day of the week and time of day in the format day hh:mm, where the day of the week is spelled out (for example, Monday).
- Monthly—Specifies the numeric date, from 1 to 31, and the time of day, in the format date hh:mm.
- Interval—Specifies the interval at which the periodic message is sent, from 1 to 60 minutes.
- Hourly—Specifies the minute of the hour at which the periodic message is sent, from 0 to 59 minutes.

Note Hourly and by interval periodic notifications are available for the Snapshot alert group only.

Message Severity Threshold

When you subscribe a destination profile to the Syslog alert group, you can set a threshold for the sending of alert group messages based on the level of severity of the message. Any message with a value lower than the destination profile specified threshold is not sent to the destination.

The severity threshold is configured using the keywords in Severity and Syslog Level Mapping and ranges from catastrophic (level 9, highest level of urgency) to debugging (level 0, lowest level of urgency).

Other alert groups do not allow setting a threshold for severity.

Note Call Home severity levels are not the same as system message logging severity levels.

Table 43: Severity and Syslog Level Mapping

Level	Keyword	Syslog Level	Description
9	**catastrophic**	—	Network-wide catastrophic failure.
8	**disaster**	—	Significant network impact.

Level	Keyword	Syslog Level	Description
7	**fatal**	Emergency (0)	System is unusable.
6	**critical**	Alert (1)	Critical conditions, immediate attention needed.
5	**major**	Critical (2)	Major conditions.
4	**minor**	Error (3)	Minor conditions.
3	**warning**	Warning (4)	Warning conditions.
2	**notification**	Notice (5)	Basic notification and informational messages. Possibly independently insignificant.
1	**normal**	Information (6)	Normal event signifying return to normal state.

Configuring Snapshot Command List

To configure the snapshot command list, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. [**no** | **default**] **alert-group-config snapshot**
4. [**no** | **default**] **add-command** *command string*
5. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters Call Home configuration submode.

	Command or Action	Purpose
Step 3	[**no** | **default**] **alert-group-config snapshot** **Example:** `Router(cfg-call-home)# alert-group-config snapshot`	Enters snapshot configuration mode. The no or default command will remove all snapshot command.
Step 4	[**no** | **default**] **add-command** *command string* **Example:** `Router(cfg-call-home-snapshot)# add-command "show version"`	Adds the command to the Snapshot alert group. The no or default command will remove the corresponding command. • *command string*—IOS command. Maximum length is 128.
Step 5	**end** **Example:** `Router(cfg-call-home-snapshot)# end`	Exits and saves the configuration.

Configuring General email Options

To use the email message transport, you must configure at least one Simple Mail Transfer Protocol (SMTP) email server address. You can configure the from and reply-to email addresses, and you can specify up to four backup email servers.

Note the following guidelines when configuring general email options:

- Backup email servers can be defined by repeating the **mail-server** command using different priority numbers.
- The **mail-server priority** *number* parameter can be configured from 1 to 100. The server with the highest priority (lowest priority number) is tried first.

To configure general email options, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. **mail-server** {[*ipv4-address* | *ipv6-address*] | *name*} **priority** *number*
4. **sender from** *email-address*
5. **sender reply-to** *email-address*
6. **source-interface** *interface-name*
7. **source-ip-address** *ipv4/ipv6 address*
8. **vrf** *vrf-name*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters Call Home configuration submode.
Step 3	**mail-server** {[*ipv4-address* | *ipv6-address*] | *name*} **priority** *number* **Example:** `Router(cfg-call-home)# mail-server stmp.example.com priority 1`	Assigns an email server address and its relative priority among configured email servers. Provide either of these: • The email server's IP address or • The email server's fully qualified domain *name* (FQDN) of 64 characters or less. Assign a priority *number* between 1 (highest priority) and 100 (lowest priority).
Step 4	**sender from** *email-address* **Example:** `Router(cfg-call-home)# sender from username@example.com`	(Optional) Assigns the email address that appears in the from field in Call Home email messages. If no address is specified, the contact email address is used.
Step 5	**sender reply-to** *email-address* **Example:** `Router(cfg-call-home)# sender reply-to username@example.com`	(Optional) Assigns the email address that appears in the reply-to field in Call Home email messages.
Step 6	**source-interface** *interface-name* **Example:** `Router(cfg-call-home)# source-interface loopback1`	Assigns the source interface name to send call-home messages. • *interface-name*—Source interface name. Maximum length is 64. **Note** For HTTP messages, use the **ip http client source-interface** *interface-name* command in global configuration mode to configure the source interface name. This allows all HTTP clients on the device to use the same source interface.

	Command or Action	Purpose
Step 7	**source-ip-address** *ipv4/ipv6 address* **Example:** `Router(cfg-call-home)# source-ip-address 209.165.200.226`	Assigns source IP address to send call-home messages. • *ipv4/ipv6 address*—Source IP (ipv4 or ipv6) address. Maximum length is 64.
Step 8	**vrf** *vrf-name* **Example:** `Router(cfg-call-home)# vrf vpn1`	(Optional) Specifies the VRF instance to send call-home email messages. If no vrf is specified, the global routing table is used. **Note** For HTTP messages, if the source interface is associated with a VRF, use the **ip http client source-interface** *interface-name* command in global configuration mode to specify the VRF instance that will be used for all HTTP clients on the device.

Example

The following example shows the configuration of general email parameters, including a primary and secondary email server:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# call-home
Router(cfg-call-home)# mail-server smtp.example.com priority 1
Router(cfg-call-home)# mail-server 192.168.0.1 priority 2
Router(cfg-call-home)# sender from username@example.com
Router(cfg-call-home)# sender reply-to username@example.com
Router(cfg-call-home)# source-interface america
Router(cfg-call-home)# source-ip-address 209.165.200.231
Router(cfg-call-home)# vrf vpn2
Router(cfg-call-home)# end
Router(config)#
```

Specifying Rate Limit for Sending Call Home Messages

To specify the rate limit for sending Call Home messages, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. **rate-limit** *number*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters Call Home configuration submode.
Step 3	**rate-limit** *number* **Example:** `Router(cfg-call-home)# rate-limit 40`	Specifies a limit on the number of messages sent per minute. • *number*—Range is 1 to 60. The default is 20.

Specifying HTTP Proxy Server

To specify an HTTP proxy server for sending Call Home HTTP(S) messages to a destination, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. **http-proxy** {*ipv4-address* | *ipv6-address* | *name*} **port** *port-number*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters Call Home configuration submode.
Step 3	**http-proxy** {*ipv4-address* | *ipv6-address* | *name*} **port** *port-number* **Example:** `Router(cfg-call-home)# http-proxy 1.1.1.1 port 1`	Specifies the proxy server for the HTTP request.

Enabling AAA Authorization to Run IOS Commands for Call Home Messages

To enable AAA authorization to run IOS commands that enable the collection of output for a Call Home message, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. **aaa-authorization**
4. **aaa-authorization** [**username** *username*]

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**call-home** **Example:**	Enters Call Home configuration submode.

	Command or Action	Purpose
	`Router(config)# call-home`	
Step 3	**aaa-authorization** **Example:** `Router(cfg-call-home)# aaa-authorization`	Enables AAA authorization. **Note** By default, AAA authorization is disabled for Call Home.
Step 4	**aaa-authorization** [**username** *username*] **Example:** `Router(cfg-call-home)# aaa-authorization username user`	Specifies the username for authorization. • **username** *username*—Default username is callhome. Maximum length is 64.

Configuring Syslog Throttling

To enable or disable Call Home syslog message throttling and avoid sending repetitive Call Home syslog messages, perform the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. [**no**] **syslog-throttling**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters Call Home configuration submode.
Step 3	[**no**] **syslog-throttling** **Example:** `Router(cfg-call-home)# syslog-throttling`	Enables or disables Call Home syslog message throttling and avoids sending repetitive Call Home syslog messages. Repeating syslog messages will only display after 24 hours. By default, syslog message throttling is enabled. **Note** Debug level syslogs like debug trace are not throttled.

Configuring Call Home Data Privacy

The **data-privacy** command scrubs data, such as IP addresses, from running configuration files to protect the privacy of customers. Enabling the **data-privacy** command can affect CPU utilization when scrubbing a large amount of data. Currently, **show** command output is not being scrubbed except for configuration messages in the **show running-config all** and **show startup-config** data.

SUMMARY STEPS

1. **configure terminal**
2. **call-home**
3. **data-privacy** {**level** {**normal** | **high**} | **hostname**}

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters configuration mode.
Step 2	**call-home** **Example:** `Router(config)# call-home`	Enters the Call Home configuration submode.
Step 3	**data-privacy** {**level** {**normal** | **high**} | **hostname**} **Example:** `Router(cfg-call-home)# data-privacy level high`	Scrubs data from running configuration file to protect the privacy of the user. The default data-privacy level is normal. **Note** Enabling the data-privacy command can affect CPU utilization when scrubbing a large amount of data. • **normal**—Scrubs all normal-level commands. • **high**—Scrubs all normal-level commands plus the IP domain name and IP address commands. • **hostname**—Scrubs all high-level commands plus the **hostname** command. **Note** Scrubbing the hostname from configuration messages can cause Smart Call Home processing failure on some platforms.

Sending Call Home Communications Manually

Sending Call Home Communications Manually

You can manually send several types of Call Home communications. To send Call Home communications, perform any necessary tasks in sections from Sending a Call Home Test Message Manually, on page 294 to Manually Sending Command Output Message for One Command or a Command List, on page 296.

Sending a Call Home Test Message Manually

You can use the **call-home test** command to send a user-defined Call Home test message.

To manually send a Call Home test message, perform the following step:

SUMMARY STEPS

1. **call-home test** [“*test-message*”] **profile** *name*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**call-home test** [“*test-message*”] **profile** *name* **Example:** `Router# call-home test profile profile1`	Sends a test message to the specified destination profile. The user-defined test message text is optional but must be enclosed in quotes (“”) if it contains spaces. If no user-defined message is configured, a default message is sent.

Sending Call Home Alert Group Messages Manually

You can use the **call-home send** command to manually send a specific alert group message.

Note the following guidelines when manually sending a Call Home alert group message:

- Only the snapshot, crash, configuration, and inventory alert groups can be sent manually. Syslog alert groups cannot be sent manually.
- When you manually trigger a snapshot, configuration, or inventory alert group message and you specify a destination profile name, a message is sent to the destination profile regardless of the profile’s active status, subscription status, or severity setting.
- When you manually trigger a snapshot, configuration, or inventory alert group message and do not specify a destination profile name, a message is sent to all active profiles that have either a normal or periodic subscription to the specified alert group.

To manually trigger Call Home alert group messages, perform the following steps:

SUMMARY STEPS

1. **call-home send alert-group snapshot** [**profile** *name*]
2. **call-home send alert-group crash** [**profile** *name*]
3. **call-home send alert-group configuration** [**profile** *name*]

4. **call-home send alert-group inventory** [**profile** *name*]

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**call-home send alert-group snapshot** [**profile** *name*] **Example:** `Router# call-home send alert-group snapshot profile profile1`	Sends a snapshot alert group message to one destination profile if specifed or to all subscribed destination profiles.
Step 2	**call-home send alert-group crash** [**profile** *name*] **Example:** `Router# call-home send alert-group crash profile profile1`	Sends a crash alert group message to one destination profile if specified or to all subscribed destination profiles.
Step 3	**call-home send alert-group configuration** [**profile** *name*] **Example:** `Router# call-home send alert-group configuration profile profile1`	Sends a configuration alert group message to one destination profile if specified or to all subscribed destination profiles.
Step 4	**call-home send alert-group inventory** [**profile** *name*] **Example:** `Router# call-home send alert-group inventory profile profile1`	Sends an inventory alert group message to one destination profile if specified or to all subscribed destination profiles.

Submitting Call Home Analysis and Report Requests

You can use the **call-home request** command to submit information about your system to Cisco to receive helpful analysis and report information specific to your system. You can request a variety of reports, including security alerts, known bugs, best practices, and command references.

Note the following guidelines when manually sending Call Home analysis and report requests:

- If a **profile** *name* is specified, the request is sent to the profile. If no profile is specified, the request is sent to the CiscoTAC-1 profile. The recipient profile does not need to be enabled for the call-home request. The profile should specify the email address where the transport gateway is configured so that the request message can be forwarded to the CiscoTAC-1 profile and the user can receive the reply from the Smart Call Home service.
- The **ccoid** *user-id* is the registered identifier of the Smart Call Home user. If the *user-id* is specified, the response is sent to the email address of the registered user. If no *user-id* is specified, the response is sent to the contact email address of the device.
- Based on the keyword specifying the type of report requested, the following information is returned:
  - **config-sanity**—Information on best practices as related to the current running configuration.

- **bugs-list**—Known bugs in the running version and in the currently applied features.
- **command-reference**—Reference links to all commands in the running configuration.
- **product-advisory**—Product Security Incident Response Team (PSIRT) notices, End of Life (EOL) or End of Sales (EOS) notices, or field notices (FN) that may affect devices in your network.

To submit a request for analysis and report information from the Cisco Output Interpreter tool, perform the following steps:

SUMMARY STEPS

1. **call-home request output-analysis "***show-command***"** [**profile** *name*] [**ccoid** *user-id*]
2. **call-home request** {**config-sanity** | **bugs-list** | **command-reference** | **product-advisory**} [**profile** *name*] [**ccoid** *user-id*]

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**call-home request output-analysis "***show-command***"** [**profile** *name*] [**ccoid** *user-id*] **Example:** `Router# call-home request output-analysis "show diag" profile TG`	Sends the output of the specified **show** command for analysis. The **show** command must be contained in quotes (“”).
Step 2	**call-home request** {**config-sanity** | **bugs-list** | **command-reference** | **product-advisory**} [**profile** *name*] [**ccoid** *user-id*] **Example:** `Router# call-home request config-sanity profile TG`	Sends the output of a predetermined set of commands such as the **show running-config all**, **show version** or **show module** commands, for analysis. In addition, the **call home request product-advisory** subcommand includes all inventory alert group commands. The keyword specified after **request** specifies the type of report requested.

Manually Sending Command Output Message for One Command or a Command List

You can use the **call-home send** command to execute an IOS command or a list of IOS commands and send the command output through HTTP or email protocol.

Note the following guidelines when sending the output of a command:

- The specified IOS command or list of IOS commands can be any run command, including commands for all modules. The command must be contained in quotes (“”).
- If the email option is selected using the “email” keyword and an email address is specified, the command output is sent to that address.
- If neither the email nor the HTTP option is specified, the output is sent in long-text format with the specified service request number to the Cisco TAC (attach@cisco.com).
- If neither the “email” nor the “http” keyword is specified, the service request number is required for both long-text and XML message formats and is provided in the subject line of the email.

- If the HTTP option is specified, the CiscoTAC-1 profile destination HTTP or HTTPS URL is used as the destination. The destination email address can be specified so that Smart Call Home can forward the message to the email address. The user must specify either the destination email address or an SR number but they can also specify both.

To execute a command and send the command output, perform the following step:

SUMMARY STEPS

1. **call-home send** {*cli command* | *cli list*} [**email** *email* **msg-format** {**long-text** | **xml**} | **http** {**destination-email-address** *email*}] [**tac-service-request** *SR#*]

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**call-home send** {*cli command* | *cli list*} [**email** *email* **msg-format** {**long-text** | **xml**} | **http** {**destination-email-address** *email*}] [**tac-service-request** *SR#*]
	Example:
	`Router# call-home send "show version;show running-config;`
	`show inventory" email support@example.com msg-format xml`
		Executes the CLI or CLI list and sends output via email or HTTP.
		• {*cli command* | *cli list*}—Specifies the IOS command or list of IOS commands (separated by ';'). It can be any run command, including commands for all modules. The commands must be contained in quotes ("").
		• **email** *email* **msg-format** {**long-text** | **xml**}—If the **email** option is selected, the command output will be sent to the specified email address in long-text or XML format with the service request number in the subject. The email address, the service request number, or both must be specified. The service request number is required if the email address is not specified (default is attach@cisco.com for long-text format and callhome@cisco.com for XML format).
		• **http** {**destination-email-address** *email*}—If the **http** option is selected, the command output will be sent to Smart Call Home backend server (URL specified in the CiscoTAC-1 profile) in XML format. **destination-email-address** *email* can be specified so that the backend server can forward the message to the email address. The email address, the service request number, or both must be specified.
		• **tac-service-request** *SR#*—Specifies the service request number. The service request number is required if the email address is not specified.

Configuring Diagnostic Signatures

Configuring Diagnostic Signatures

The Diagnostic Signatures feature downloads digitally signed signatures to devices. Diagnostic Signatures (DS) files are formatted files that collate knowledge of diagnostic events and provide methods to troubleshoot them without a need to upgrade the Cisco software. The aim of DS is to deliver flexible intelligence that can detect and collect troubleshooting information that can be used to resolve known problems in customers networks.

Prerequisites for Diagnostic Signatures

Before you download and configure diagnostic signatures (DSs) on a device, you must ensure that the following conditions are met:

- You must assign one or more DSs to the device. See the Diagnostic Signature Downloading, on page 299 for more information on how to assign DSs to devices.
- HTTP/Secure HTTP (HTTPS) transport is required for downloading DS files. You must install the certification authority (CA) certificate to enable the authentication of the destination HTTPS server.

Note If you configure the trustpool feature, the CA certificate is not required.

Information About Diagnostic Signatures

Diagnostic Signatures Overview

Diagnostic signatures (DS) for the Call Home system provides a flexible framework that allows the defining of new events and corresponding CLIs that can analyze these events without upgrading the Cisco software.

DSs provide the ability to define more types of events and trigger types than the standard Call Home feature supports. The DS subsystem downloads and processes files on a device as well as handles callbacks for diagnostic signature events.

The Diagnostic Signature feature downloads digitally signed signatures that are in the form of files to devices. DS files are formatted files that collate the knowledge of diagnostic events and provide methods to troubleshoot these events.

DS files contain XML data to specify the event description, and these files include CLI commands or scripts to perform required actions. These files are digitally signed by Cisco or a third party to certify their integrity, reliability, and security.

The structure of a DS file can be one of the following formats:

- Metadata-based simple signature that specifies the event type and contains other information that can be used to match the event and perform actions such as collecting information by using the CLI. The signature can also change configurations on the device as a workaround for certain bugs.

- Embedded Event Manager (EEM) Tool Command Language (Tcl) script-based signature that specifies new events in the event register line and additional action in the Tcl script.
- Combination of both the formats above.

The following basic information is contained in a DS file:

- ID (unique number): unique key that represents a DS file that can be used to search a DS.
- Name (ShortDescription): unique description of the DS file that can be used in lists for selection.
- Description: long description about the signature.
- Revision: version number, which increments when the DS content is updated.
- Event & Action: defines the event to be detected and the action to be performed after the event happens.

Diagnostic Signature Downloading

To download the diagnostic signature (DS) file, you require the secure HTTP (HTTPS) protocol. If you have already configured an email transport method to download files on your device, you must change your assigned profile transport method to HTTPS to download and use DS.

Cisco software uses a PKI Trustpool Management feature, which is enabled by default on devices, to create a scheme to provision, store, and manage a pool of certificates from known certification authorities (CAs). The trustpool feature installs the CA certificate automatically. The CA certificate is required for the authentication of the destination HTTPS servers.

There are two types of DS update requests to download DS files: regular and forced-download. Regular download requests DS files that were recently updated. You can trigger a regular download request either by using a periodic configuration or by initiating an on-demand CLI. The regular download update happens only when the version of the requested DS is different from the version of the DS on the device. Periodic download is only started after there is any DS assigned to the device from DS web portal. After the assignment happens, the response to the periodic inventory message from the same device will include a field to notify device to start its periodic DS download/update. In a DS update request message, the status and revision number of the DS is included such that only a DS with the latest revision number is downloaded.

Forced-download downloads a specific DS or a set of DSes. You can trigger the forced-download update request only by initiating an on-demand CLI. In a force-download update request, the latest version of the DS file is downloaded irrespective of the current DS file version on the device.

The DS file is digitally signed, and signature verification is performed on every downloaded DS file to make sure it is from a trusted source.

Diagnostic Signature Workflow

The diagnostic signature feature is enabled by default in Cisco software. The following is the workflow for using diagnostic signatures:

1. Find the DS(es) you want to download and assign them to the device. This step is mandatory for regular periodic download, but not required for forced download.
2. The device downloads all assigned DS(es) or a specific DS by regular periodic download or by on-demand forced download.
3. The device verifies the digital signature of every single DS. If verification passes, the device stores the DS file into a non-removable disk, such as bootflash or hard disk. This is so that DS files can be read after the device is reloaded. For example, on the Cisco CSR 1000v, the DS file is stored in the bootflash:/call home directory.

4. The device continues sending periodic regular DS download requests to get the latest revision of DS and replace the older one in device.
5. The device monitors the event and executes the actions defined in the DS when the event happens.

Diagnostic Signature Events and Actions

The events and actions sections are the key areas used in diagnostic signatures. The event section defines all event attributes that are used for event detection. The action section lists all actions which should be performed after the event happens, such as collecting s**how** command outputs and sending them to Smart Call Home to parse.

Diagnostic Signature Event Detection

Event detection in a DS is defined in two ways: single event detection and multiple event detection.

Diagnostic Signature Actions

The diagnostic signature (DS) file consists of various actions that must be initiated when an event occurs. The action type indicates the kind of action that will be initiated in response to a certain event.

Variables are elements within a DS that are used to customize the files.

DS actions are categorized into the following four types:

- call-home
- command
- emailto
- script

DS action types call-home and emailto collect event data and send a message to call-home servers or to the defined email addresses. The message uses “diagnostic-signature” as its message type and DS ID as the message sub-type.

The commands defined for the DS action type initiate CLI commands that can change configuration of the device, collect show command outputs, or run any EXEC command on the device. The DS action type script executes Tcl scripts.

Diagnostic Signature Variables

Variables are referenced within a DS and are used to customize the DS file. All DS variable names have the prefix ds_ to separate them from other variables. The following are the supported DS variable types:

- System variable: variables assigned automatically by the device without any configuration changes. The Diagnostic Signatures feature supports two system variables: ds_hostname and ds_signature_id.
- Environment variable: values assigned manually by using the **environment** *variable-name variable-value* command in call-home diagnostic-signature configuration mode. Use the **show call-home diagnostic-signature** command to display the name and value of all DS environment variables. If the DS file contains unresolved environment variables, this DS will stay in pending status until the variable gets resolved.
- Prompt variable: values assigned manually by using the **call-home diagnostic-signature install** *ds-id* command in privileged EXEC mode. If you do not set this value, the status of the DS indicates pending.
- Regular expression variable: values assigned from a regular expression pattern match with predefined CLI command outputs. The value is assigned during the DS run.

- Syslog event variable: values assigned during a syslog event detection in the DS file. This variable is valid only for syslog event detection.

How to Configure Diagnostic Signatures

Configuring the Call Home Service for Diagnostic Signatures

Configure the Call Home Service feature to set attributes such as the contact email address where notifications related with diagnostic signatures (DS) are sent and destination HTTP/secure HTTP (HTTPS) URL to download the DS files from.

You can also create a new user profile, configure correct attributes and assign it as the DS profile. For periodic downloads, the request is sent out just following full inventory message. By changing the inventory periodic configuration, the DS periodic download also gets rescheduled.

Note The predefined CiscoTAC-1 profile is enabled as a DS profile by default and we recommend using it. If used, you only need to change the destination transport-method to the **http** setting.

SUMMARY STEPS

1. **configure terminal**
2. **service call-home**
3. **call-home**
4. **contact-email-addr** *email-address*
5. **mail-server** {*ipv4-addr* | *name*} **priority** *number*
6. **profile** *profile-name*
7. **destination transport-method** {**email** | **http**}
8. **destination address** {**email** *address* | **http** *url*}
9. **subscribe-to-alert-group inventory** [**periodic** {**daily** *hh*:*mm* | **monthly** *day hh*:*mm* | **weekly** *day hh*:*mm*}]
10. **exit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**service call-home** **Example:**	Enables Call Home service on a device.

	Command or Action	Purpose
	`Router(config)# service call-home`	
Step 3	**call-home** **Example:** `Router(config)# call-home`	Enters call-home configuration mode for the configuration of Call Home settings.
Step 4	**contact-email-addr** *email-address* **Example:** `Router(cfg-call-home)# contact-email-addr userid@example.com`	(Optional) Assigns an email address to be used for Call Home customer contact.
Step 5	**mail-server** {*ipv4-addr* | *name*} **priority** *number* **Example:** `Router(cfg-call-home)# mail-server 10.1.1.1 priority 4`	(Optional) Configures a Simple Mail Transfer Protocol (SMTP) email server address for Call Home. This command is only used when sending email is part of the actions defined in any DS.
Step 6	**profile** *profile-name* **Example:** `Router(cfg-call-home)# profile user1`	Configures a destination profile for Call Home and enters call-home profile configuration mode.
Step 7	**destination transport-method** {**email** | **http**} **Example:** `Router(cfg-call-home-profile)# destination transport-method http`	Specifies a transport method for a destination profile in the Call Home. **Note** To configure diagnostic signatures, you must use the **http** option.
Step 8	**destination address** {**email** *address* | **http** *url*} **Example:** `Router(cfg-call-home-profile)# destination address http <address>`	Configures the address type and location to which call-home messages are sent. **Note** To configure diagnostic signatures, you must use the **http** option.
Step 9	**subscribe-to-alert-group inventory** [**periodic** {**daily** *hh*:*mm* | **monthly** *day hh*:*mm* | **weekly** *day hh*:*mm*}] **Example:** `Router(cfg-call-home-profile)# subscribe-to-alert-group inventory periodic daily 14:30`	Configures a destination profile to send messages for the Inventory alert group for Call Home. • This command is used only for the periodic downloading of DS files.
Step 10	**exit** **Example:** `Router(cfg-call-home-profile)# exit`	Exits call-home profile configuration mode and returns to call-home configuration mode.

What to do next

What to Do Next

Set the profile configured in the previous procedure as the DS profile and configure other DS parameters.

Configuring Diagnostic Signatures

The Diagnostic Signatures feature downloads digitally signed signatures to devices. Diagnostic Signatures (DS) files are formatted files that collate knowledge of diagnostic events and provide methods to troubleshoot them without a need to upgrade the Cisco software. The aim of DS is to deliver flexible intelligence that can detect and collect troubleshooting information that can be used to resolve known problems in customers networks.

Configuration Examples for Diagnostic Signatures

The following example shows how to enable the periodic downloading request for diagnostic signature (DS) files. This configuration will send download requests to the service call-home server daily at 2:30 p.m. to check for updated DS files. The transport method is set to HTTP.

```
Router> enable
Router# configure terminal
Router(config)# service call-home
Router(config)# call-home
Router(cfg-call-home)# contact-email-addr userid@example.com
Router(cfg-call-home)# mail-server 10.1.1.1 priority 4
Router(cfg-call-home)# profile user-1
Router(cfg-call-home-profile)# destination transport-method http
Router(cfg-call-home-profile)# destination address http
https://tools.cisco.com/its/service/oddce/services/DDCEService
Router(cfg-call-home-profile)# subscribe-to-alert-group inventory periodic daily 14:30
Router(cfg-call-home-profile)# exit
Router(cfg-call-home)# diagnostic-signature
Router(cfg-call-home-diag-sign)# profile user1
Router(cfg-call-home-diag-sign)# environment ds_env1 envarval
Router(cfg-call-home-diag-sign)# end
```

The following is sample output from the **show call-home diagnostic-signature** command for the configuration displayed above:

```
Router# show call-home diagnostic-signature
Current diagnostic-signature settings:
Diagnostic-signature: enabled
Profile: user1 (status: ACTIVE)
Environment variable:
ds_env1: abc
Downloaded DSes:
DS ID    DS Name                         Revision Status     Last Update (GMT+00:00)
-------- ------------------------------- -------- ---------- -------------------
6015     CronInterval                    1.0      registered 2013-01-16 04:49:52
6030     ActCH                           1.0      registered 2013-01-16 06:10:22
6032     MultiEvents                     1.0      registered 2013-01-16 06:10:37
6033     PureTCL                         1.0      registered 2013-01-16 06:11:48
```

Displaying Call Home Configuration Information

You can use variations of the **show call-home** command to display Call Home configuration information.

SUMMARY STEPS

1. **show call-home**
2. **show call-home detail**
3. **show call-home alert-group**
4. **show call-home mail-server status**
5. **show call-home profile** {**all** | *name*}
6. **show call-home statistics** [**detail** | **profile** *profile-name*]

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**show call-home** **Example:** `Router# show call-home`	Displays the Call Home configuration in summary.
Step 2	**show call-home detail** **Example:** `Router# show call-home detail`	Displays the Call Home configuration in detail.
Step 3	**show call-home alert-group** **Example:** `Router# show call-home alert-group`	Displays the available alert groups and their status.
Step 4	**show call-home mail-server status** **Example:** `Router# show call-home mail-server status`	Checks and displays the availability of the configured email server(s).
Step 5	**show call-home profile** {**all**	*name*} **Example:** `Router# show call-home profile all`	Displays the configuration of the specified destination profile. Use the **all** keyword to display the configuration of all destination profiles.
Step 6	**show call-home statistics** [**detail**	**profile** *profile-name*] **Example:** `Router# show call-home statistics`	Displays the statistics of Call Home events.

Examples

Examples 1 to 7 show sample output when using different options of the **show call-home** command.

Call Home Information in Summary

```
Router# show call-home
Current call home settings:
    call home feature : enable
    call home message's from address: router@example.com
    call home message's reply-to address: support@example.com
    vrf for call-home messages: Not yet set up
    contact person's email address: technical@example.com
    contact person's phone number: +1-408-555-1234
    street address: 1234 Picaboo Street, Any city, Any state, 12345
    customer ID: ExampleCorp
    contract ID: X123456789
    site ID: SantaClara
    source ip address: Not yet set up
    source interface: GigabitEthernet1
    Mail-server[1]: Address: 192.168.2.1 Priority: 1
    Mail-server[2]: Address: 209.165.254.254 Priority: 2
    http proxy: 192.168.1.1:80
    aaa-authorization: disable
    aaa-authorization username: callhome (default)
    data-privacy: normal
    syslog throttling: enable
    Rate-limit: 20 message(s) per minute
    Snapshot command[0]: show version
    Snapshot command[1]: show clock
Available alert groups:
    Keyword                  State   Description
    ------------------------ ------- --------------------------------
    configuration            Enable  configuration info
    crash                    Enable  crash and traceback info
    inventory                Enable  inventory info
    snapshot                 Enable  snapshot info
    syslog                   Enable  syslog info
Profiles:
    Profile Name: campus-noc
    Profile Name: CiscoTAC-1
```

Call Home Information in Detail

```
Router# show call-home detail

Current call home settings:
    call home feature : enable
    call home message's from address: router@example.com
    call home message's reply-to address: support@example.com
    vrf for call-home messages: Not yet set up
    contact person's email address: technical@example.com
    contact person's phone number: +1-408-555-1234
    street address: 1234 Picaboo Street, Any city, Any state, 12345
    customer ID: ExampleCorp
    contract ID: X123456789
    site ID: SantaClara
    source ip address: Not yet set up
    source interface: GigabitEthernet1
```

```
    Mail-server[1]: Address: 192.168.2.1 Priority: 1
    Mail-server[2]: Address: 209.165.254.254 Priority: 2
    http proxy: 192.168.1.1:80
    aaa-authorization: disable
    aaa-authorization username: callhome (default)
    data-privacy: normal
    syslog throttling: enable
    Rate-limit: 20 message(s) per minute
    Snapshot command[0]: show version
    Snapshot command[1]: show clock
Available alert groups:
    Keyword                  State   Description
    ------------------------ ------- --------------------------------
    configuration            Enable  configuration info
    crash                    Enable  crash and traceback info
    inventory                Enable  inventory info
    snapshot                 Enable  snapshot info
    syslog                   Enable  syslog info
Profiles:
Profile Name: campus-noc
    Profile status: ACTIVE
    Preferred Message Format: xml
    Message Size Limit: 3145728 Bytes
    Transport Method: email
    Email address(es): noc@example.com
    HTTP  address(es): Not yet set up
    Alert-group               Severity
    ------------------------  ------------
    configuration             normal
    crash                     normal
    inventory                 normal
    Syslog-Pattern            Severity
    ------------------------  ------------
 .*CALL_LOOP.*             debug
Profile Name: CiscoTAC-1
    Profile status: INACTIVE
    Profile mode: Full Reporting
    Preferred Message Format: xml
    Message Size Limit: 3145728 Bytes
    Transport Method: email
    Email address(es): callhome@cisco.com
    HTTP  address(es): https://tools.cisco.com/its/service/oddce/services/DDCEService
    Periodic configuration info message is scheduled every 14 day of the month at 11:12
    Periodic inventory info message is scheduled every 14 day of the month at 10:57
    Alert-group               Severity
    ------------------------  ------------
    crash                     normal

    Syslog-Pattern            Severity
    ------------------------  ------------
 .*CALL_LOOP.*             debug
```

Available Call Home Alert Groups

```
Router# show call-home alert-group

Available alert groups:
    Keyword                  State   Description
    ------------------------ ------- --------------------------------
    configuration            Enable  configuration info
    crash                    Enable  crash and traceback info
    inventory                Enable  inventory info
```

```
    snapshot                Enable  snapshot info
    syslog                  Enable  syslog info
```

email Server Status Information

```
Router# show call-home mail-server status
Please wait. Checking for mail server status ...
    Mail-server[1]: Address: 192.168.2.1 Priority: 1 [Not Available]
    Mail-server[2]: Address: 209.165.254.254 Priority: 2 [Available]
```

Information for All Destination Profiles

```
Router# show call-home profile all
Profile Name: campus-noc
    Profile status: ACTIVE
    Preferred Message Format: xml
    Message Size Limit: 3145728 Bytes
    Transport Method: email
    Email address(es): noc@example.com
    HTTP  address(es): Not yet set up
    Alert-group               Severity
    ------------------------  ------------
    configuration             normal
    crash                     normal
    inventory                 normal
    Syslog-Pattern            Severity
    ------------------------  ------------
 .*CALL_LOOP.*             debug
Profile Name: CiscoTAC-1
    Profile status: INACTIVE
    Profile mode: Full Reporting
    Preferred Message Format: xml
    Message Size Limit: 3145728 Bytes
    Transport Method: email
    Email address(es): callhome@cisco.com
    HTTP  address(es): https://tools.cisco.com/its/service/oddce/services/DDCEService
    Periodic configuration info message is scheduled every 14 day of the month at 11:12
    Periodic inventory info message is scheduled every 14 day of the month at 10:57
    Alert-group               Severity
    ------------------------  ------------
    crash                     normal

    Syslog-Pattern            Severity
    ------------------------  ------------
 .*CALL_LOOP.*             debug
```

Information for a User-Defined Destination Profile

```
Router# show call-home profile campus-noc
Profile Name: campus-noc
    Profile status: ACTIVE
    Preferred Message Format: xml
    Message Size Limit: 3145728 Bytes
    Transport Method: email
    Email address(es): noc@example.com
    HTTP  address(es): Not yet set up
    Alert-group               Severity
    ------------------------  ------------
    configuration             normal
    crash                     normal
```

```
    inventory                normal
    Syslog-Pattern           Severity
    -----------------------  ------------
 .*CALL_LOOP.*             debug
```

Call Home Statistics

```
Router# show call-home statistics
Message Types    Total                Email                HTTP
-------------   -------------------- -------------------- ------------------
Total Success   3                    3                    0
    Config      3                    3                    0
    Crash       0                    0                    0
    Inventory   0                    0                    0
    Snapshot    0                    0                    0
    SysLog      0                    0                    0
    Test        0                    0                    0
    Request     0                    0                    0
    Send-CLI    0                    0                    0
Total In-Queue  0                    0                    0
    Config      0                    0                    0
    Crash       0                    0                    0
    Inventory   0                    0                    0
    Snapshot    0                    0                    0
    SysLog      0                    0                    0
    Test        0                    0                    0
    Request     0                    0                    0
    Send-CLI    0                    0                    0
Total Failed    0                    0                    0
    Config      0                    0                    0
    Crash       0                    0                    0
    Inventory   0                    0                    0
    Snapshot    0                    0                    0
    SysLog      0                    0                    0
    Test        0                    0                    0
    Request     0                    0                    0
    Send-CLI    0                    0                    0
Total Ratelimit
      -dropped  0                    0                    0
    Config      0                    0                    0
    Crash       0                    0                    0
    Inventory   0                    0                    0
    Snapshot    0                    0                    0
    SysLog      0                    0                    0
    Test        0                    0                    0
    Request     0                    0                    0
    Send-CLI    0                    0                    0
Last call-home message sent time: 2011-09-26 23:26:50 GMT-08:00
```

Default Settings

Table 44: Default Call Home Settings

Parameters	Default
Call Home feature status	Disabled
User-defined profile status	Active

Parameters	Default
Predefined CiscoTAC-1 profile status	Inactive
Transport method	email
Message format type	XML
Alert group status	Enabled
Call Home message severity threshold	Debug
Message rate limit for messages per minute	20
AAA authorization	Disabled
Call Home syslog message throttling	Enabled
Data privacy level	Normal

Alert Group Trigger Events and Commands

Call Home trigger events are grouped into alert groups, with each alert group assigned commands to execute when an event occurs. The command output is included in the transmitted message. The *Call Home Alert Groups, Events, and Actions* section lists the trigger events included in each alert group, including the severity level of each event and the executed commands for the alert group.

Table 45: Call Home Alert Groups, Events, and Actions

Alert Group	Call Home Trigger Event	Syslog Event	Severity	Description and Commands Executed
Crash	SYSTEM_CRASH	—	—	Events related to system crash. Commands executed: **show version**, **show logging**, **show region**, **show stack**.
—	TRACEBACK	—	—	Detects software traceback events. Commands executed: **show version**, **show logging**, **show region**, **show stack**.
Configuration	—	—	—	User-generated request for configuration or configuration change event. Commands executed: **show platform**,**show running-config all**, **show startup-config**, **show version**.

Alert Group	Call Home Trigger Event	Syslog Event	Severity	Description and Commands Executed
Inventory	—	—	—	User-generated request for inventory event. Commands executed: **show diag all eeprom detail | include MAC**, **show license all** , **show platform**, **show platform hardware qfp active infrastructure chipset 0 capabilities**, **show platform software vnic-if interface-mapping**, **show version**.
Syslog	—	—	—	User-generated Syslog event. Commands executed: **show logging**

Message Contents

The following tables display the content formats of alert group messages:

- The section and table below, "Format for a Short Text Message", shows the content fields of a short text message.
- The section and table below, "Common Fields for All Long Text and XML Messages", shows the content fields that are common to all long text and XML messages. The fields specific to a particular alert group message are inserted at a point between the common fields. The insertion point is identified in the table.

To see a sample syslog message alert, see Sample Syslog Alert Notification in XML Format, on page 314.

Table 46: Format for a Short Text Message

Data Item	Description
Device identification	Configured device name
Date/time stamp	Time stamp of the triggering event
Error isolation message	Plain English description of triggering event
Alarm urgency level	Error level such as that applied to a system message

Table 47: Common Fields for All Long Text and XML Messages

Data Item(Plain Text and XML)	Description(Plain Text and XML)	Call-Home Message Tag (XML Only)
Time stamp	Date and time stamp of event in ISO time notation: *YYYY-MM-DD HH:MM:SS GMT+HH:MM*.	CallHome/EventTime
Message name	Name of message. Specific event names are listed in Alert Group Trigger Events and Commands, on page 309.	For short text message only
Message type	Specifically “Call Home”.	CallHome/Event/Type
Message subtype	Specific type of message: full, delta, test	CallHome/Event/SubType
Message group	Specifically “reactive”. Optional because default is “reactive”.	For long-text message only
Severity level	Severity level of message (see table "Severity and Syslog Level Mapping" in section Message Severity Threshold, on page 285).	Body/Block/Severity
Source ID	Product type for routing through the workflow engine. This is typically the product family name.	For long-text message only

Data Item(Plain Text and XML)	Description(Plain Text and XML)	Call-Home Message Tag (XML Only)
Device ID	Unique device identifier (UDI) for end device generating message. This field should be empty if the message is nonspecific to a fabric switch. The format is *type@Sid@seria* l. • *type* is the product model number from backplane IDPROM. • **@** is a separator character. • *Sid* is C, identifying the serial ID as a chassis serial number. • *serial* is the number identified by the Sid field. Example: `CISCO3845@C@12345678`	CallHome/CustomerData/ContractData/DeviceId
Customer ID	Optional user-configurable field used for contract information or other ID by any support service.	CallHome/CustomerData/ContractData/CustomerId
Contract ID	Optional user-configurable field used for contract information or other ID by any support service.	CallHome/CustomerData/ContractData/ContractId
Site ID	Optional user-configurable field used for Cisco-supplied site ID or other data meaningful to alternate support service.	CallHome/CustomerData/ContractData/SiteId

Data Item(Plain Text and XML)	Description(Plain Text and XML)	Call-Home Message Tag (XML Only)
Server ID	If the message is generated from the fabric switch, this is the unique device identifier (UDI) of the switch. The format is *type@Sid@serial*. • *type* is the product model number from backplane IDPROM. • **@** is a separator character. • *Sid* is C, identifying the serial ID as a chassis serial number. • *serial* is the number identified by the Sid field. Example: `CISCO3845@C@12345678`	For long text message only
Message description	Short text describing the error.	CallHome/MessageDescription
Device name	Node that experienced the event. This is the host name of the device.	CallHome/CustomerData/SystemInfo/NameName
Contact name	Name of person to contact for issues associated with the node experiencing the event.	CallHome/CustomerData/SystemInfo/Contact
Contact email	email address of person identified as contact for this unit.	CallHome/CustomerData/SystemInfo/ContactEmail
Contact phone number	Phone number of the person identified as the contact for this unit.	CallHome/CustomerData/SystemInfo/ContactPhoneNumber
Street address	Optional field containing street address for RMA part shipments associated with this unit.	CallHome/CustomerData/SystemInfo/StreetAddress

Data Item(Plain Text and XML)	Description(Plain Text and XML)	Call-Home Message Tag (XML Only)
Model name	Model name of the router. This is the “specific model as part of a product family name.	CallHome/Device/Cisco_Chassis/Model
Serial number	Chassis serial number of the unit.	CallHome/Device/Cisco_Chassis/SerialNumber
System object ID	System Object ID that uniquely identifies the system.	`CallHome/Device/Cisco_Chassis/AdditionalInformation/AD@name=“sysObjectID”`
System description	System description for the managed element.	`CallHome/Device/Cisco_Chassis/AdditionalInformation/AD@name=“sysDescr”`

Table 48: Inserted Fields Specific to a Particular Alert Group Message

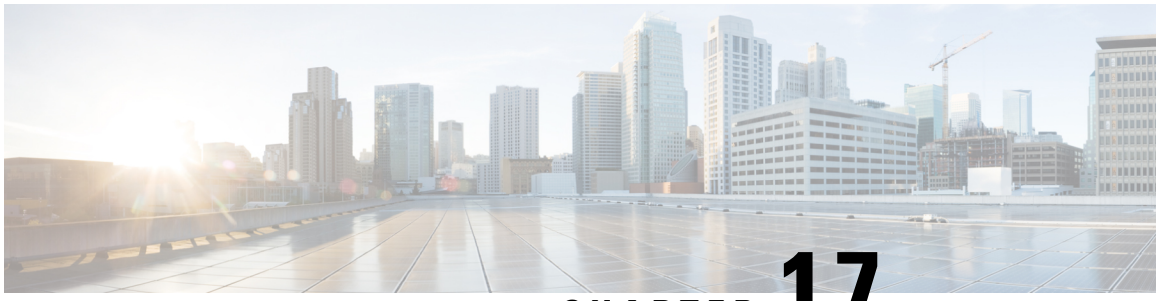
Data Item(Plain Text and XML)	Description(Plain Text and XML)	Call-Home Message Tag (XML Only)
The following fields may be repeated if multiple commands are executed for this alert group.		
Command output name	Exact name of the issued command.	/aml/Attachments/Attachment/Name
Attachment type	Attachment type. Usually “inline”.	/aml/Attachments/Attachment@type
MIME type	Normally “text” or “plain” or encoding type.	/aml/Attachments/Attachment/Data@encoding
Command output text	Output of command automatically executed (see table "Call Home Alert Groups, Events, and Actions" in section Alert Group Trigger Events and Commands, on page 309).	/mml/attachments/attachment/atdata

Sample Syslog Alert Notification in XML Format

```
<?xml version="1.0" encoding="UTF-8"?>
<soap-env:Envelope xmlns:soap-env="http://www.w3.org/2003/05/soap-envelope">
<soap-env:Header>
<aml-session:Session xmlns:aml-session="http://www.cisco.com/2004/01/aml-session"
soap-env:mustUnderstand="true"
soap-env:role="http://www.w3.org/2003/05/soap-envelope/role/next">
<aml-session:To>http://tools.cisco.com/neddce/services/DDCEService</aml-session:To>
<aml-session:Path>
<aml-session:Via>http://www.cisco.com/appliance/uri</aml-session:Via>
</aml-session:Path>
<aml-session:From>http://www.cisco.com/appliance/uri</aml-session:From>
<aml-session:MessageId>M8:9S1NMSF22DW:51AEAC68</aml-session:MessageId>
</aml-session:Session>
</soap-env:Header>
<soap-env:Body>
```

```
<aml-block:Block xmlns:aml-block="http://www.cisco.com/2004/01/aml-block">
<aml-block:Header>
<aml-block:Type>http://www.cisco.com/2005/05/callhome/syslog</aml-block:Type>
<aml-block:CreationDate>2013-06-05 03:11:36 GMT+00:00</aml-block:CreationDate>
<aml-block:Builder>
<aml-block:Name>CSR1000v</aml-block:Name>
<aml-block:Version>2.0</aml-block:Version>
</aml-block:Builder>
<aml-block:BlockGroup>
<aml-block:GroupId>G9:9S1NMSF22DW:51AEAC68</aml-block:GroupId>
<aml-block:Number>0</aml-block:Number>
<aml-block:IsLast>true</aml-block:IsLast>
<aml-block:IsPrimary>true</aml-block:IsPrimary>
<aml-block:WaitForPrimary>false</aml-block:WaitForPrimary>
</aml-block:BlockGroup>
<aml-block:Severity>2</aml-block:Severity>
</aml-block:Header>
<aml-block:Content>
<ch:CallHome xmlns:ch="http://www.cisco.com/2005/05/callhome" version="1.0">
<ch:EventTime>2013-06-05 03:11:36 GMT+00:00</ch:EventTime> <ch:MessageDescription>*Jun  5
03:11:36.041: %CLEAR-5-COUNTERS: Clear counter on all interfaces by
console</ch:MessageDescription> <ch:Event> <ch:Type>syslog</ch:Type> <ch:SubType></ch:SubType>
 <ch:Brand>Cisco Systems</ch:Brand> <ch:Series>CSR1000v Cloud Services Router</ch:Series>
</ch:Event> <ch:CustomerData> <ch:UserData> <ch:Email>weijuhua@cisco.com</ch:Email>
</ch:UserData>
<ch:ContractData>
<ch:CustomerId></ch:CustomerId>
<ch:SiteId></ch:SiteId>
<ch:ContractId></ch:ContractId>
<ch:DeviceId>CSR1000v@C@9S1NMSF22DW</ch:DeviceId>
</ch:ContractData>
<ch:SystemInfo>
<ch:Name>qiang-vm</ch:Name>
<ch:Contact></ch:Contact>
<ch:ContactEmail>weijuhua@cisco.com</ch:ContactEmail>
<ch:ContactPhoneNumber></ch:ContactPhoneNumber>
<ch:StreetAddress></ch:StreetAddress>
</ch:SystemInfo>
<ch:CCOID></ch:CCOID>
<ch:IdToken></ch:IdToken>
</ch:CustomerData>
<ch:Device>
<rme:Chassis xmlns:rme="http://www.cisco.com/rme/4.0">
<rme:Model>CSR1000v</rme:Model>
<rme:HardwareVersion></rme:HardwareVersion>
<rme:SerialNumber>9S1NMSF22DW</rme:SerialNumber>
<rme:AdditionalInformation>
<rme:AD name="PartNumber" value="" />
<rme:AD name="SoftwareVersion" value="15.4(20130604:093915)" /> <rme:AD name="SystemObjectId"
 value="1.3.6.1.4.1.9.1.1537" /> <rme:AD name="SystemDescription" value="Cisco IOS Software,
 CSR1000v Software (X86_64_LINUX_IOSD-ADVENTERPRISEK9-M), Experimental Version
15.4(20130604:093915) [mcp_dev-qiazhou-ultra_ut 100] Copyright (c) 1986-2013 by Cisco
Systems, Inc.
Compiled Tue 04-Jun-13 02:39 by jsmith" /> <rme:AD name="ServiceNumber" value="" /> <rme:AD
 name="ForwardAddress" value="" /> </rme:AdditionalInformation> </rme:Chassis> </ch:Device>
 </ch:CallHome> </aml-block:Content> <aml-block:Attachments> <aml-block:Attachment
type="inline"> <aml-block:Name>show logging</aml-block:Name> <aml-block:Data encoding="plain">
 <![CDATA[show logging Syslog logging: enabled (0 messages dropped, 1 messages rate-limited,
 0 flushes, 0 overruns, xml disabled, filtering disabled)
No Active Message Discriminator.
No Inactive Message Discriminator.
    Console logging: level debugging, 391 messages logged, xml disabled,
                     filtering disabled
    Monitor logging: level debugging, 0 messages logged, xml disabled,
```

```
                  filtering disabled
    Buffer logging:  level debugging, 391 messages logged, xml disabled,
                   filtering disabled
    Exception Logging: size (4096 bytes)
    Count and timestamp logging messages: disabled
    Persistent logging: disabled
No active filter modules.
    Trap logging: level informational, 56 message lines logged
        Logging Source-Interface:       VRF Name:
Log Buffer (4096 bytes):
*Jun  5 03:11:18.295: %SYS-5-CONFIG_I: Configured from console by console
qiang-vm#]]></aml-block:Data> </aml-block:Attachment> </aml-block:Attachments>
</aml-block:Block> </soap-env:Body> </soap-env:Envelope>
```



CHAPTER 17

Enabling Management by REST API

Introduction

You can use the Cisco IOS XE REST API to manage the Cisco CSR 1000v as an alternative to configuring and managing selected features on the router using the Cisco IOS XE CLI. This chapter describes how to configure the Cisco CSR 1000v to enable management using the REST API. For detailed information about using the REST API, see Cisco IOS XE REST API Management Reference Guide.

Note REST API is not supported from the IOS-XE 16.7.x release onwards. If you are using the 16.7.x version or above, Cisco recommends that you use Restconf. For more information on using Restconf, see the Restconf documentation.

Enabling REST API Support During Cisco CSR 1000v OVA Deployment

If you are deploying the Cisco CSR 1000v OVA template, support for REST API is configured in the Bootstrap Properties screen of the OVA Wizard. The required fields are different depending on the Cisco IOS XE release. The tables below list the fields required to enable REST API support when deploying the OVA template.

For more information on deploying the OVA template, see *the Deploying the Cisco CSR 1000v OVA to the VM* section in this guide.

Table 49: Cisco CSR 1000v OVA Template Bootstrap Properties Required for REST API Support (Cisco IOS XE Release 3.12S and Later)

Property	Description
Management Interface	Designates the management interface for the Cisco CSR 1000v. The format must be GigabitEthernetx or GigabitEthernetx.xxx.

Property	Description
Management Interface IPv4 Address/Mask	Configures the IPv4 address and subnet mask for the management interface.
Management IPv4 Gateway (Cisco IOS XE Release 3.12S)	Configures the IPv4 management default gateway address. If using DHCP, enter “dhcp” in the field.
Management IPv4 Network (Cisco IOS XE Release 3.12S)	Configures the IPv4 Network (such as “192.168.2.0/24” or “192.168.2.0 255.255.255.0”) that the management gateway should route to. If a default route (0.0.0.0/0) is desired, this may be left blank.

Table 50: Cisco CSR 1000v OVA Template Bootstrap Properties Required for REST API Support (Cisco IOS XE Release 3.11S and Later)

Property	Description
Management Interface	Designates the management interface for the Cisco CSR 1000v. The format must be GigabitEthernetx or GigabitEthernetx.xxx.
Management Interface IPv4 Address/Mask	Configures the IPv4 address and subnet mask for the management interface.
Management IPv4 Default Gateway	Configures the IPv4 management default gateway address. If using DHCP, enter “dhcp” in the field.
Remote Management IPv4 Address(not used if configuring the shared management interface to support REST API).	Configures the IP address used for remote management of the Cisco CSR 1000v by the REST API or by Cisco PNSC. The address must be in the same subnet as the management interface address.

Table 51: Cisco CSR 1000v OVA Template Bootstrap Properties Required for REST API Support (Cisco IOS XE Release 3.10S)

Property	Description
Management IPv4 Address/Mask	Sets the management gateway address and mask in IPv4 format for the GigabitEthernet0 management interface.
Management IPv4 Default Gateway	Sets the default management gateway IP address in IPv4 format for the GigabitEthernet0 management interface. **Note** The GigabitEthernet0 interface is no longer supported beginning in Cisco IOS XE Release 3.11S.
Enable HTTPS Server	Enables an HTTPS server for system configuration and administration via a web browser. Required if using the REST API to perform system management in Cisco IOS XE Release 3.10S.

Enabling REST API Support Using the Cisco IOS XE CLI

Introduction to REST API Configuration Options

You need to configure the management interface to support REST API using the Cisco IOS XE CLI if you installed the Cisco CSR 1000v in either of the following ways:

- If you installed the Cisco CSR 1000v using the .iso file.
- If you deployed the Cisco CSR 1000v using an Amazon Machine Image (AMI).

Note If upgrading a REST API configuration from Cisco IOS XE Release 3.10S to a later release, you must add your REST API configuration to the IOS configuration.

Before configuring the shared management interface, perform the steps in Enabling REST API Support, on page 319.

The REST API management is located in a management virtual services container that is separate from the router components, including the router management interface. You have two choices for configuring the REST API management support, and the steps for each of these are in the following two sections:

- Configuring the Shared Management Interface to Support the REST API , on page 320

  (Cisco IOS XE 3.13S and later, and IOS XE Denali 16.3 and later) The router management interface and the virtual services management container can share the same IP address. This can be used to save an IP address to be allocated for other purposes.

- Configuring the Dual Management Interface to Support the REST API , on page 322

  (Required in Cisco IOS XE 3.11S and 3.12S, optional in later releases.) The router management interface and the virtual services management container use different IP addresses.

The remainder of this section contains information about:

- Configuring the REST API Local Port and AutoSave Options, on page 324
- Configuring HTTPS Support for the REST API Using the Cisco IOS XE CLI, on page 325
- Disabling REST API Support, on page 327
- Viewing the REST API Container Status, on page 328

Enabling REST API Support

Beginning with Cisco IOS XE Release 3.11S, and including IOS XE Denali 16.3.1 and later, you can enable REST API support on the remote management interface. To disable REST API support, see Disabling REST API Support, on page 327. To enable the REST API, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **remote-management**
4. **restful-api**
5. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `router# configure terminal`	Enters global configuration mode.
Step 3	**remote-management** **Example:** `router(config)# remote-management`	Enters remote-management configuration mode.
Step 4	**restful-api** **Example:** `router(cfg-remote-mgmt)# restful-api`	Enables support for the REST API.
Step 5	**end** **Example:** `router(cfg-remote-mgmt)# end`	Exits remote-management configuration mode and enters configuration mode.

Configuring the Shared Management Interface to Support the REST API

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *mgmt-interface*
4. **ip address** *mgmt-ipv4-addr subnet-mask*

5. **no shutdown**
6. **exit**
7. **virtual-service csr_mgmt**
8. **no activate**
9. **ip shared host-interface** *mgmt-interface*
10. **activate**
11. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *mgmt-interface* **Example:** `Router(config)# interface gigabitethernet1`	Enters interface configuration mode for the management interface.
Step 4	**ip address** *mgmt-ipv4-addr subnet-mask* **Example:** `Router(config-if)# ip address 172.25.29.235 255.255.255.128`	Configures the IP address for the management interface.
Step 5	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Enables the management interface.
Step 6	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode.
Step 7	**virtual-service csr_mgmt** **Example:**	Configures the **csr_mgmt** virtual services container and enters virtual services configuration mode.

	Command or Action	Purpose
	`router(config)# virtual-service csr_mgmt`	
Step 8	**no activate** **Example:** `router(config-virt-serv)# no activate`	Deactivates the **csr_mgmt** virtual services container.
Step 9	**ip shared host-interface** *mgmt-interface* **Example:** `router(config-virt-serv)# ip shared host-interface gigabitethernet 1`	Maps the virtual service container to the management interface.
Step 10	**activate** **Example:** `router(config-virt-serv)# activate`	Activates the **csr_mgmt** virtual services container.
Step 11	**end** **Example:** `router(config-virt-serv)# end`	Exits virtual services configuration mode and enters global configuration mode.

Configuring the Dual Management Interface to Support the REST API

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface GigabitEthernet***x*
4. **ip address** *ipv4-addr subnet-mask*
5. **no shutdown**
6. **exit**
7. **interface virtualportgroup** *virtualportgroup-number*
8. **ip unnumbered GigabitEthernet***x*
9. **no shutdown**
10. **exit**
11. **virtual-service csr_mgmt**
12. **vnic gateway virtualportgroup** *virtualportgroup_number*
13. **guest ip address** *remote-mgmt-ipv4-addr*
14. **exit**
15. **end**
16. **ip route** *ipaddress subnetmask* **virtualportgroup** *virtualportgroupnumber*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface GigabitEthernet***x* **Example:** `Router(config)# interface gigabitethernet1`	Enters interface configuration mode for the interface designated by *x*. The range of GigabitEthernet ports depends on the platform.
Step 4	**ip address** *ipv4-addr subnet-mask* **Example:** `Router(config-if)# ip address 172.25.29.235 255.255.255.128`	Configures the IP address for the management interface.
Step 5	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Enables the management interface.
Step 6	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode.
Step 7	**interface virtualportgroup** *virtualportgroup-number* **Example:** `Router(config)# interface virtualportgroup 0`	Creates a virtual port group and enters virtual port group interface configuration mode.
Step 8	**ip unnumbered GigabitEthernet***x* **Example:** `router(config-if)# ip unnumbered gigabitethernet1`	Enables IP processing on an interface without assigning it an explicit IP address.
Step 9	**no shutdown** **Example:**	Enables the virtual port group interface.

	Command or Action	Purpose
	`router(config-if)# no shutdown`	
Step 10	**exit** **Example:** `router(config-if)# exit`	Exits virtual port group interface mode.
Step 11	**virtual-service csr_mgmt** **Example:** `router(config)# virtual-service csr_mgmt`	Configures the **csr_mgmt** virtual services container and enters virtual services configuration mode.
Step 12	**vnic gateway virtualportgroup** *virtualportgroup_number* **Example:** `router(config-virt-serv)# vnic gateway virtualportgroup 0`	Creates a vNIC gateway interface for the virtual services container and maps it to the virtual port group.
Step 13	**guest ip address** *remote-mgmt-ipv4-addr* **Example:** `router(config-virt-serv-intf)# guest ip address 172.25.29.236`	Configures the remote-management IP address for the vNIC gateway interface for the virtual services container.
Step 14	**exit** **Example:** `router(config-virt-serv-intf)# exit`	Exits virtual services interface configuration mode and enters virtual services configuration mode.
Step 15	**end** **Example:** `router(config-virt-serv)# end`	Exits virtual services configuration mode and enters global configuration mode.
Step 16	**ip route** *ipaddress subnetmask* **virtualportgroup** *virtualportgroupnumber* **Example:** `router(config)# ip route 172.25.29.236 255.255.255.255 VirtualPortGroup0`	Creates an IP route that maps to the virtual port group. Use the same IP address that was configured using the **guest ip address** command.

Configuring the REST API Local Port and AutoSave Options

Beginning with Cisco IOS XE 3.13S, you can configure the REST API local port and autosave options.

SUMMARY STEPS

1. **remote-management**
2. **restful-api local-port** *local-port-number*
3. **restful-api autosave** *interval*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**remote-management** **Example:** `router(config)# remote-management`	Enters remote-management configuration mode.
Step 2	**restful-api local-port** *local-port-number* **Example:** `router(cfg-remote-mgmt)# restful-api local-port 55443`	Configures the REST API local port number. The valid range depends on whether the REST API virtual services container uses the same IP address as the management interface, or if it uses a different IP address: • Valid range if the dual management interface is configured is from 1 to 61000. • Valid range if the shared management interface is configured is from 55001 to 61000. In both cases, the default value is 55443.
Step 3	**restful-api autosave** *interval* **Example:** `Router(cfg-remote-mgmt)# restful-api autosave 60`	Configures the REST API autosave interval. The range is from 30-300 seconds, and the default is 30.

Configuring HTTPS Support for the REST API Using the Cisco IOS XE CLI

The REST API requires HTTPS server support. Beginning with Cisco IOS XE Release 3.11S, HTTPS server support is enabled by default and no additional configuration is required. However, if using Cisco IOS XE Release 3.10S, you must manually configure HTTPS support for the REST API in the following situations:

- If you did not specify the Enable HTTPS Server option when deploying the OVA.
- If you installed the Cisco CSR 1000v using the .iso file.

Note The HTTPS session must have an identity certificate. For more information, see the “HTTPS-HTTP Server and Client with SSL 3.0” section of the HTTP Services Configuration Guide, Cisco IOS XE Release 3S.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip http secure-server**
4. **transport-map type persistent webui** *transport-map-name*
5. **secure-server**
6. **transport type persistent webui input** *transport-map-name*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `router# configure terminal`	Enters global configuration mode.
Step 3	**ip http secure-server** **Example:** `router(config)# ip http secure-server`	Enables HTTPS on port 443 (the default HTTPS port). A self-signed identity certificate is automatically generated.
Step 4	**transport-map type persistent webui** *transport-map-name* **Example:** `router(config)# transport-map type persistent webui https-webui`	Creates and names a persistent web user interface transport map.
Step 5	**secure-server** **Example:** `router(config)# secure-server`	Enables the secure HTTPS server.
Step 6	**transport type persistent webui input** *transport-map-name* **Example:** `router(config)# transport type persistent webui input https-webui`	Enables the transport map to support HTTPS.

Disabling REST API Support

Beginning with Cisco IOS XE Release 3.11S, and including IOS XE Denali 16.3.1 and later, you can disable REST API support on the remote management interface. To enable REST API support, see Enabling REST API Support, on page 319. To disable the REST API, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **remote-management**
4. **no restful-api**
5. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `router# configure terminal`	Enters global configuration mode.
Step 3	**remote-management** **Example:** `router(config)# remote-management`	Enters remote-management configuration mode.
Step 4	**no restful-api** **Example:** `router(cfg-remote-mgmt)# no restful-api`	Disables support for the REST API.
Step 5	**end** **Example:** `router(cfg-remote-mgmt)# end`	Exits remote-management configuration mode and enters configuration mode.

What to do next

Note When REST API support is disabled using the **no restful-api** command, the REST API PUT, POST and DELETE operations are disabled. However, the GET operation is still available.

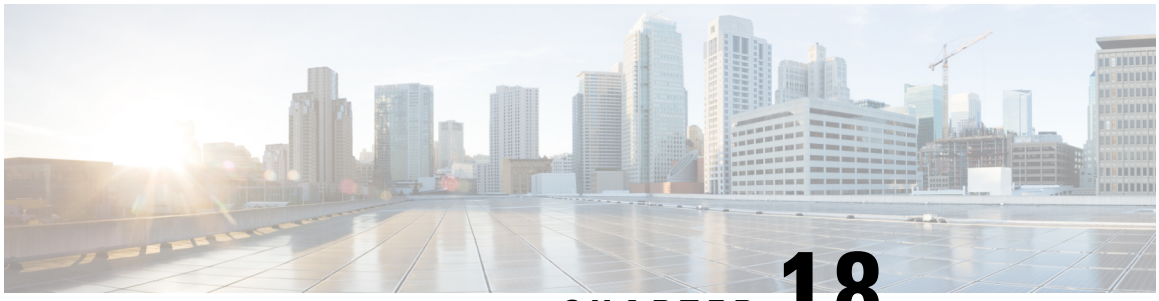
Viewing the REST API Container Status

Use the **show virtual-service detail** command to view the REST API container status.

The following example shows the enabled status of the REST API container, along with the detailed guest status with a list of processes, status showing when these processes are up and running, and the number of restarts:

```
Router# show virtual-service detail
Virtual service csr_mgmt detail
  State                : Activated
  Package information
    Name               : csrmgmt.1_2_1.20131010_134115.ova
    Path               : bootflash:/csrmgmt.1_2_1.20131010_134115.ova
    Application
      Name             : csr_mgmt
      Installed version : 1.2.1
      Description      : CSR-MGMT
    Signing
      Key type         : Cisco development key
      Method           : SHA-1
    Licensing
      Name             : Not Available
      Version          : Not Available
  Detailed guest status

------------------------------------------------------------------------
Process               Status            Uptime           # of restarts
------------------------------------------------------------------------
nginx                 UP         0Y 0W 0D  0: 1: 1        0
climgr                UP         0Y 0W 0D  0: 1: 1        0
restful_api           UP         0Y 0W 0D  0: 1: 1        0
fcgicpa               UP         0Y 0W 0D  0: 0:13        0
pnscag                UP         0Y 0W 0D  0: 0:13        0
pnscdme               UP         0Y 0W 0D  0: 0:12        0
------------------------------------------------------------------------
Feature         Status                Configuration
------------------------------------------------------------------------
Restful API   Enabled, UP             port: 443
              (GET only)              auto-save-timer: 8 seconds
                                      socket: unix:/usr/local/nginx/csrapi-fcgi.sock;
PNSC          Enabled, UP             host: 172.25.223.233
                                      port:  8443
                                      socket: unix:/usr/local/cpa-fcgi.sock;
Network stats:
 eth0: RX  packets:38, TX  packets:6
 eth1: RX  packets:87, TX  packets:80
Coredump file(s):
  Activated profile name: None
  Resource reservation
    Disk               : 540 MB
    Memory             : 512 MB
    CPU                : 30% system CPU
```

```
  Attached devices
    Type              Name        Alias
    ------------------------------------------
    Serial/Trace                  serial3
    Serial/Syslog                 serial2
    Serial/aux                    serial1
    Serial/shell                  serial0
    Disk              /opt/var
    Disk              _rootfs
    NIC               dp_2_0      net2
    NIC               ieobc_2     ieobc
  Network interfaces
    MAC address             Attached to interface
    ------------------------------------------------------
    00:1E:BD:DE:F8:BA       VirtualPortGroup0
    54:0E:00:0B:0C:03       ieobc_2
  Guest interface
  ---
  Interface: eth1
  ip address: 172.25.223.147/25
  ---
  Guest routes
  ---
  Address/Mask                          Next Hop                            Intf.
------------------------------------------------------------------------------------
0.0.0.0/0                             172.25.223.137                      eth1
  ---
  Resource admission (without profile) : passed
    Disk space    : 540MB
    Memory        : 512MB
    CPU           : 30% system CPU
    VCPUs         : Not specified
```



CHAPTER 18

Radio Aware Routing

Radio Aware Routing

Radio-Aware Routing (RAR) is a mechanism that uses radios to interact with the routing protocol OSPFv3 to signal the appearance, disappearance, and link conditions of one-hop routing neighbors.

In a large mobile networks, connections to the routing neighbors are often interrupted due to distance and radio obstructions. When these signals do not reach the routing protocols, protocol timers are used to update the status of a neighbor. Routing protocols have lengthy timer, which is not recommended in mobile networks.

The RAR feature is supported on Cisco ISR G2 and G3 Series Routers, Cisco ISR 4000 Series Routers.

PPPoE Extensions is the RAR protocol supported in Cisco 4000 Series ISRs. PPPoE Extensions with Aggregate support is introduce from Cisco IOS XE Fuji 16.7. release. OSPFv3 and EIGRP are the supported routing protocols.

Benefits of Radio Aware Routing

The Radio Aware Routing feature offers the following benefits:

- Provides faster network convergence through immediate recognition of changes.
- Enables routing for failing or fading radio links.
- Allows easy routing between line-of-sight and non-line-of-sight paths.
- Provides faster convergence and optimal route selection so that delay-sensitive traffic, such as voice and video, is not disrupted
- Provides efficient radio resources and bandwidth usage.
- Reduces impact on the radio links by performing congestion control in the router.
- Allows route selection based on radio power conservation.

- Enables decoupling of the routing and radio functionalities.
- Provides simple Ethernet connection to RFC 5578, R2CP, and DLEP compliant radios.

Restrictions and Limitations

The Radio Aware Routing feature has the following restrictions and limitations:

- The DLEP and R2CP protocols are not supported in Cisco 4000 Series ISRs.
- Multicast traffic is not supported in aggregate mode.
- Cisco High Availability (HA) technology is not supported.

License Requirements

This feature is available with the AX license.

Performance

The Radio Aware Routing feature has the ability to support a maximum of 10 neighbors per radio or VMI interface; and a total of 30 to 40 neighbors.

System Components

The Radio Aware Routing (RAR) feature is implemented using the MANET (Mobile adhoc network) infrastructure comprising of different components such as PPPoE, Virtual multipoint interface (VMI), QoS, routing protocol interface and RAR protocols.

Point-to-Point Protocol over Ethernet PPPoE or PPPoE

PPPoE is a well-defined communication mechanism between the client and the server. In the RAR implementation, radio takes the role of the PPPoE client and router takes the role of the PPPoE server. This allows a loose coupling of radio and router, while providing a well-defined and predictable communication mechanism.

As PPPoE is a session or a connection oriented protocol, it extends the point-to-point radio frequency (RF) link from an external radio to an IOS router.

PPPoE Extensions

PPPoE extensions are used when the router communicates with the radio. In the Cisco IOS implementation of PPPoE, each individual session is represented by virtual access interface (connectivity to a radio neighbor) on which, QoS can be applied with these PPPoE extensions.

RFC5578 provides extensions to PPPoE to support credit-based flow control and session-based real time link metrics, which are very useful for connections with variable bandwidth and limited buffering capabilities (such as radio links).

Virtual Multipoint Interface (VMI)

Though PPPoE Extensions provides the most of the setup to communicate between a router and a radio, VMI addresses the need to manage and translate events that higher layers (example, routing protocols) consume. In addition, VMI operates in the Bypass mode.

In Bypass mode, every Virtual Access Interface (VAI) representing a radio neighbor is exposed to routing protocols OSPFv3 and EIGRP, so that, the routing protocol directly communicates with the respective VAI for both unicast and multicast routing protocol traffic.

In Aggregae mode, VMI is exposed to the routing protocols (OSPF) so that the routing protocols can leverage VMI for their optimum efficiency. When the network neighbors are viewed as a collection of networks on a point-to-multipoint link with broadcast and multicast capability at VMI, VMI helps in aggregating the multiple virtual access interfaces created from PPPoE. VMI presents a single multi access layer 2 broadcast capable interface. The VMI layer handles re-directs unicast routing protocol traffic to the appropriate P2P link (Virtual-Access interface), and replicats any Multicast/Broadcast traffic that needs to flow. Since the routing protocol communicates to a single interface, the size of the topology database is reduced, without impacting the integrity of the network.

QoS Provisioning on PPPoE Extension Session

The following example describes QoS provisioning on PPPoE extension session:

```
policy-map rar_policer
 class class-default
  police 10000 2000 1000 conform-action transmit  exceed-action drop  violate-action drop
policy-map rar_shaper
 class class-default
  shape average percent 1

interface Virtual-Template2
 ip address 10.92.2.1 255.255.255.0
 no peer default ip address
 no keepalive
 service-policy input rar_policer
end
```

Example: Configuring the RAR Feature in Bypass Mode

The following example is an end-to-end configuration of RAR in the bypass mode:

Note Before you being the RAR configuration, you must first configure the **subscriber authorization enable** command to bring up the RAR session. Without enbaling authorization, the Point-to-Point protocol does not recognize this as a RAR session and may not tag *manet_radio* in presentation of a PPPoE Active Discovery Initiate (PADI). By default, bypass mode does not appears in the configuration. It appears only if the mode is configured as bypass.

Configure a Service for RAR

```
policy-map type service rar-lab
 pppoe service manet_radio //note: Enter the pppoe service policy name as manet_radio
!
```

Configure Broadband

```
bba-group pppoe VMI2
 virtual-template 2
service profile rar-lab
!
interface GigabitEthernet0/0/0
 description Connected to Client1
  negotiation auto
  pppoe enable group VMI2
!
```

Configure a Service for RAR

```
policy-map type service rar-lab
 pppoe service manet_radio //note: Enter the pppoe service policy name as manet_radio
!
```

Configuration in Bypass Mode

- IP Address Configured under Virtual-Template Explicitly

```
interface Virtual-Template2
ip address 192.0.2.3 255.255.255.0
no ip redirects
peer default ip address pool PPPoEpool2
ipv6 enable
ospfv3 1 network manet
ospfv3 1 ipv4 area 0
ospfv3 1 ipv6 area 0
no keepalive
service-policy input rar_policer Or/And
service-policy output rar_shaper
```

- VMI Unnumbered Configured under Virtual Template

```
interface Virtual-Template2
ip unnumbered vmi2
no ip redirects
peer default ip address pool PPPoEpool2
ipv6 enable
ospfv3 1 network manet
ospfv3 1 ipv4 area 0
ospfv3 1 ipv6 area 0
no keepalive
service-policy input rar_policer Or/And
service-policy output rar_shaper
```

Configure the Virtual Multipoint Interface in Bypass Mode

```
interface vmi2 //configure the virtual multi interface
ip address 192.0.2.1 255.255.255.0
```

```
physical-interface GigabitEthernet0/0/0
mode bypass

interface vmi3//configure the virtual multi interface
 ip address 192.0.2.3 255.255.255.0
 physical-interface GigabitEthernet0/0/1
mode bypass
```

Configure OSPF Routing

```
router ospfv3 1
 router-id 192.0.2.1
!
 address-family ipv4 unicast
  redistribute connected metric 1 metric-type 1
  log-adjacency-changes
 exit-address-family
 !
 address-family ipv6 unicast
  redistribute connected metric-type 1
  log-adjacency-changes
 exit-address-family
!
ip local pool PPPoEpool2 198.51.100.1 198.51.100.254
```

Verifying RAR Session Details

To retrieve RAR session details, use the following show commands:

```
Router#show pppoe session packets all
Total PPPoE sessions 2

session id: 9
local MAC address: 006b.f10e.a5e0, remote MAC address: 0050.56bc.424a
virtual access interface: Vi2.1, outgoing interface: Gi0/0/0
    1646 packets sent, 2439363 received
    176216 bytes sent, 117250290 received

PPPoE Flow Control Stats
 Local Credits: 65535   Peer Credits: 65535   Local Scaling Value 64 bytes
 Credit Grant Threshold: 28000    Max Credits per grant: 65535
 Credit Starved Packets: 0
 PADG xmit Seq Num: 32928     PADG Timer index: 0
 PADG last rcvd Seq Num: 17313
 PADG last nonzero Seq Num: 17306
 PADG last nonzero rcvd amount: 2
 PADG Timers: (ms)   [0]-1000    [1]-2000    [2]-3000    [3]-4000    [4]-5000
 PADG xmit: 33308  rcvd: 17313
 PADC xmit: 17313  rcvd: 19709
 In-band credit pkt xmit: 7 rcvd: 2434422
 Last credit packet snapshot
  PADG xmit: seq_num = 32928, fcn = 0, bcn = 65535
  PADC rcvd: seq_num = 32928, fcn = 65535, bcn = 65535
  PADG rcvd: seq_num = 17313, fcn = 0, bcn = 65535
  PADC xmit: seq_num = 17313, fcn = 65535, bcn = 65535
  In-band credit pkt xmit: fcn = 61, bcn = 65533
  In-band credit pkt rcvd: fcn = 0, bcn = 65534
    ==== PADQ Statistics ====
     PADQ xmit: 0  rcvd: 0
```

```
session id: 10
local MAC address: 006b.f10e.a5e1, remote MAC address: 0050.56bc.7dcb
virtual access interface: Vi2.2, outgoing interface: Gi0/0/1
    1389302 packets sent, 1852 received
    77869522 bytes sent, 142156 received

PPPoE Flow Control Stats
 Local Credits: 65535   Peer Credits: 65535   Local Scaling Value 64 bytes
 Credit Grant Threshold: 28000    Max Credits per grant: 65535
 Credit Starved Packets: 0
 PADG xmit Seq Num: 18787     PADG Timer index: 0
 PADG last rcvd Seq Num: 18784
 PADG last nonzero Seq Num: 18768
 PADG last nonzero rcvd amount: 2
 PADG Timers: (ms)   [0]-1000    [1]-2000    [2]-3000    [3]-4000    [4]-5000
 PADG xmit: 18787  rcvd: 18784
 PADC xmit: 18784  rcvd: 18787
 In-band credit pkt xmit: 1387764 rcvd: 956
 Last credit packet snapshot
  PADG xmit: seq_num = 18787, fcn = 0, bcn = 65535
  PADC rcvd: seq_num = 18787, fcn = 65535, bcn = 65535
  PADG rcvd: seq_num = 18784, fcn = 0, bcn = 65535
  PADC xmit: seq_num = 18784, fcn = 65535, bcn = 65535
  In-band credit pkt xmit: fcn = 0, bcn = 64222
  In-band credit pkt rcvd: fcn = 0, bcn = 65534
    ==== PADQ Statistics ====
     PADQ xmit: 0  rcvd: 1


Router#show pppoe session packets
Total PPPoE sessions 2

SID     Pkts-In         Pkts-Out        Bytes-In        Bytes-Out
9       2439391         1651            117252098       176714
10      1858            1389306         142580          77869914


Router#show vmi counters
Interface vmi2: - Last Clear Time =

Input Counts:
  Process Enqueue      =          0 (VMI)
  Fastswitch           =          0
  VMI Punt Drop:
       Queue Full      =          0

Output Counts:
  Transmit:
       VMI Process DQ  =       4280
       Fastswitch VA   =          0
       Fastswitch VMI  =          0
  Drops:
       Total           =          0
       QOS Error       =          0
       VMI State Error =          0
       Mcast NBR Error =          0
       Ucast NBR Error =          0
Interface vmi3: - Last Clear Time =

Input Counts:
  Process Enqueue      =          0 (VMI)
  Fastswitch           =          0
  VMI Punt Drop:
```

```
       Queue Full      =          0

Output Counts:
  Transmit:
       VMI Process DQ  =       2956
       Fastswitch VA   =          0
       Fastswitch VMI  =          0
  Drops:
       Total           =          0
       QOS Error       =          0
       VMI State Error =          0
       Mcast NBR Error =          0
       Ucast NBR Error =          0
Interface vmi4: - Last Clear Time =

Input Counts:
  Process Enqueue      =          0 (VMI)
  Fastswitch           =          0
  VMI Punt Drop:
       Queue Full      =          0

Output Counts:
  Transmit:
       VMI Process DQ  =          0
       Fastswitch VA   =          0
       Fastswitch VMI  =          0
  Drops:
       Total           =          0
       QOS Error       =          0
       VMI State Error =          0
       Mcast NBR Error =          0
       Ucast NBR Error =          0
Router#


Router#show vmi neighbor details
1 vmi2 Neighbors
      1 vmi3 Neighbors
      0 vmi4 Neighbors
      2 Total Neighbors

vmi2   IPV6 Address=FE80::21E:E6FF:FE43:F500
       IPV6 Global Addr=::
       IPV4 Address=192.0.2.2, Uptime=05:15:01
       Output pkts=89, Input pkts=0
       No Session Metrics have been received for this neighbor.
       Transport PPPoE, Session ID=9
       INTERFACE STATS:
          VMI Interface=vmi2,
             Input qcount=0, drops=0, Output qcount=0, drops=0
          V-Access intf=Virtual-Access2.1,
             Input qcount=0, drops=0, Output qcount=0, drops=0
          Physical intf=GigabitEthernet0/0/0,
             Input qcount=0, drops=0, Output qcount=0, drops=0

PPPoE Flow Control Stats
 Local Credits: 65535   Peer Credits: 65535   Local Scaling Value 64 bytes
 Credit Grant Threshold: 28000    Max Credits per grant: 65535
 Credit Starved Packets: 0
 PADG xmit Seq Num: 33038     PADG Timer index: 0
 PADG last rcvd Seq Num: 17423
 PADG last nonzero Seq Num: 17420
 PADG last nonzero rcvd amount: 2
 PADG Timers: (ms)   [0]-1000    [1]-2000    [2]-3000    [3]-4000    [4]-5000
```

```
 PADG xmit: 33418  rcvd: 17423
 PADC xmit: 17423  rcvd: 19819
 In-band credit pkt xmit: 7 rcvd: 2434446
 Last credit packet snapshot
  PADG xmit: seq_num = 33038, fcn = 0, bcn = 65535
  PADC rcvd: seq_num = 33038, fcn = 65535, bcn = 65535
  PADG rcvd: seq_num = 17423, fcn = 0, bcn = 65535
  PADC xmit: seq_num = 17423, fcn = 65535, bcn = 65535
  In-band credit pkt xmit: fcn = 61, bcn = 65533
  In-band credit pkt rcvd: fcn = 0, bcn = 65534
    ==== PADQ Statistics ====
     PADQ xmit: 0  rcvd: 0


vmi3   IPV6 Address=FE80::21E:7AFF:FE68:6100
       IPV6 Global Addr=::
       IPV4 Address=192.0.2.4, Uptime=05:14:55
       Output pkts=6, Input pkts=0
       METRIC DATA: Total rcvd=1, Avg arrival rate (ms)=0
          CURRENT: MDR=128000 bps, CDR=128000 bps
                   Lat=0 ms, Res=100, RLQ=100, load=0
          MDR      Max=128000 bps, Min=128000 bps, Avg=128000 bps
          CDR      Max=128000 bps, Min=128000 bps, Avg=128000 bps
          Latency  Max=0, Min=0, Avg=0 (ms)
          Resource Max=100%, Min=100%, Avg=100%
          RLQ      Max=100, Min=100, Avg=100
          Load     Max=0%, Min=0%, Avg=0%
       Transport PPPoE, Session ID=10
       INTERFACE STATS:
          VMI Interface=vmi3,
             Input qcount=0, drops=0, Output qcount=0, drops=0
          V-Access intf=Virtual-Access2.2,
             Input qcount=0, drops=0, Output qcount=0, drops=0
          Physical intf=GigabitEthernet0/0/1,
             Input qcount=0, drops=0, Output qcount=0, drops=0

PPPoE Flow Control Stats
 Local Credits: 65535   Peer Credits: 65535   Local Scaling Value 64 bytes
 Credit Grant Threshold: 28000    Max Credits per grant: 65535
 Credit Starved Packets: 0
 PADG xmit Seq Num: 18896     PADG Timer index: 0
 PADG last rcvd Seq Num: 18894
 PADG last nonzero Seq Num: 18884
 PADG last nonzero rcvd amount: 2
 PADG Timers: (ms)   [0]-1000    [1]-2000    [2]-3000    [3]-4000    [4]-5000
 PADG xmit: 18896  rcvd: 18894
 PADC xmit: 18894  rcvd: 18896
 In-band credit pkt xmit: 1387764 rcvd: 961
 Last credit packet snapshot
  PADG xmit: seq_num = 18896, fcn = 0, bcn = 65535
  PADC rcvd: seq_num = 18896, fcn = 65535, bcn = 65535
  PADG rcvd: seq_num = 18894, fcn = 0, bcn = 65535
  PADC xmit: seq_num = 18894, fcn = 65535, bcn = 65535
  In-band credit pkt xmit: fcn = 0, bcn = 64222
  In-band credit pkt rcvd: fcn = 0, bcn = 65534
    ==== PADQ Statistics ====
     PADQ xmit: 0  rcvd: 1


Router#show vmi neighbor details vmi 2
              1 vmi2 Neighbors

vmi2   IPV6 Address=FE80::21E:E6FF:FE43:F500
       IPV6 Global Addr=::
```

```
       IPV4 Address=192.0.2.2, Uptime=05:16:03
       Output pkts=89, Input pkts=0
       No Session Metrics have been received for this neighbor.
       Transport PPPoE, Session ID=9
       INTERFACE STATS:
          VMI Interface=vmi2,
             Input qcount=0, drops=0, Output qcount=0, drops=0
          V-Access intf=Virtual-Access2.1,
             Input qcount=0, drops=0, Output qcount=0, drops=0
          Physical intf=GigabitEthernet0/0/0,
             Input qcount=0, drops=0, Output qcount=0, drops=0

PPPoE Flow Control Stats
 Local Credits: 65535   Peer Credits: 65535   Local Scaling Value 64 bytes
 Credit Grant Threshold: 28000    Max Credits per grant: 65535
 Credit Starved Packets: 0
 PADG xmit Seq Num: 33100     PADG Timer index: 0
 PADG last rcvd Seq Num: 17485
 PADG last nonzero Seq Num: 17449
 PADG last nonzero rcvd amount: 2
 PADG Timers: (ms)   [0]-1000    [1]-2000    [2]-3000    [3]-4000    [4]-5000
 PADG xmit: 33480  rcvd: 17485
 PADC xmit: 17485  rcvd: 19881
 In-band credit pkt xmit: 7 rcvd: 2434460
 Last credit packet snapshot
  PADG xmit: seq_num = 33100, fcn = 0, bcn = 65535
  PADC rcvd: seq_num = 33100, fcn = 65535, bcn = 65535
  PADG rcvd: seq_num = 17485, fcn = 0, bcn = 65535
  PADC xmit: seq_num = 17485, fcn = 65535, bcn = 65535
  In-band credit pkt xmit: fcn = 61, bcn = 65533
  In-band credit pkt rcvd: fcn = 0, bcn = 65534
    ==== PADQ Statistics ====
     PADQ xmit: 0  rcvd: 0


Router#show platform hardware qfp active feature ess session
Current number sessions: 2
Current number TC flow: 0
Feature Type: A=Accounting D=Policing(DRL) F=FFR M=DSCP Marking L=L4redirect P=Portbundle
T=TC

   Session    Type      Segment1      SegType1      Segment2      SegType2 Feature Other
-------------------------------------------------------------------------------------------
        21     PPP 0x0000001500001022    PPPOE 0x0000001500002023    LTERM -------
        24     PPP 0x0000001800003026    PPPOE 0x0000001800004027    LTERM -------


Router#show platform software subscriber pppoe_fctl evsi 21
PPPoE Flow Control Stats
 Local Credits: 65535   Peer Credits: 65535   Local Scaling Value 64 bytes
 Credit Grant Threshold: 28000    Max Credits per grant: 65535
 Credit Starved Packets: 0
 PADG xmit Seq Num: 33215     PADG Timer index: 0
 PADG last rcvd Seq Num: 17600
 PADG last nonzero Seq Num: 17554
 PADG last nonzero rcvd amount: 2
 PADG Timers: (ms)   [0]-1000    [1]-2000    [2]-3000    [3]-4000    [4]-5000
 PADG xmit: 33595  rcvd: 17600
 PADC xmit: 17600  rcvd: 19996
 In-band credit pkt xmit: 7 rcvd: 2434485
 Last credit packet snapshot
  PADG xmit: seq_num = 33215, fcn = 0, bcn = 65535
  PADC rcvd: seq_num = 33215, fcn = 65535, bcn = 65535
  PADG rcvd: seq_num = 17600, fcn = 0, bcn = 65535
```

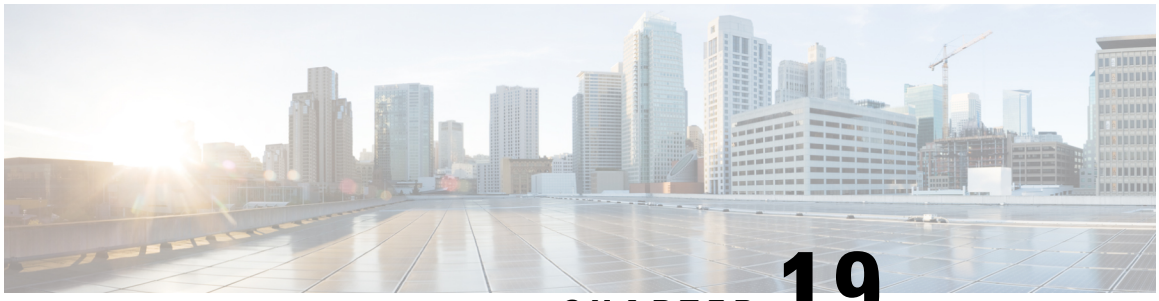
```
  PADC xmit: seq_num = 17600, fcn = 65535, bcn = 65535
  In-band credit pkt xmit: fcn = 61, bcn = 65533
  In-band credit pkt rcvd: fcn = 0, bcn = 65534

BQS buffer statistics
 Current packets in BQS buffer: 0
 Total en-queue packets: 0 de-queue packets: 0
 Total dropped packets: 0

Internal flags: 0x0
```

```
Router#show platform hardware qfp active feature ess session id 21
Session ID: 21

  EVSI type: PPP
  SIP Segment ID: 0x1500001022
  SIP Segment type: PPPOE
  FSP Segment ID: 0x1500002023
  FSP Segment type: LTERM
  QFP if handle: 16
  QFP interface name: EVSI21
  SIP TX Seq num: 0
  SIP RX Seq num: 0
  FSP TX Seq num: 0
  FSP RX Seq num: 0
  Condition Debug: 0x00000000
    session
```

```
Router#show ospfv3 neighbor

          OSPFv3 1 address-family ipv4 (router-id 10.3.3.3)

Neighbor ID     Pri   State           Dead Time   Interface ID    Interface
192.0.2.1         0    FULL/  -        00:01:32    19              Virtual-Access2.1

          OSPFv3 1 address-family ipv6 (router-id 10.3.3.3)

Neighbor ID     Pri   State           Dead Time   Interface ID    Interface
192.0.2.1         0    FULL/  -        00:01:52    19              Virtual-Access2.1
Router#
```

```
Router#sh ip route

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
a - application route
+ - replicated route, % - next hop override, p - overrides from PfR

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
C        10.90.90.0/24 is directly connected, Virtual-Access2.1
O        10.90.90.4/32 [110/1] via 192.0.2.4, 00:00:03, Virtual-Access2.1
L        10.90.90.5/32 is directly connected, Virtual-Access2.1
```

```
      10.92.90.0/32 is subnetted, 1 subnets
C        10.92.2.21 is directly connected, Virtual-Access2.1
```



CHAPTER 19

Configuring Support for Remote Management by the Cisco Prime Network Services Controller

Configuring the Management Interface to Support Remote Management by the Cisco Prime Network Services Controller

Note The Cisco Prime Network Services Controller is unsupported using Cisco IOS XE Denali 16.3.1 or later, on the Cisco CSR 1000v.

(Cisco IOS XE Denali 16.3 or earlier) You can use the Cisco Prime Network Services Controller to provision, manage and monitor the Cisco CSR 1000v. This procedure configures the Cisco CSR 1000v management interface to support remote management using the Cisco Prime Network Services Controller.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *mgmt-interface*
4. **ip address** *mgmt-ipv4-addr subnet-mask*
5. **no shutdown**
6. **exit**
7. **interface virtualportgroup** *virtual-port-group-number-number*
8. **ip unnumbered** *management-interface*
9. **no shutdown**
10. **exit**
11. **virtual-service csr_mgmt**
12. **vnic gateway virtualportgroup** *virtual-port-group-number*

13. **guest ip address** *remote-mgmt-ipv4-addr*
14. **exit**
15. **activate**
16. **end**
17. **ip route** *ip-address subnet-mask* **virtualportgroup** *virtual-port-group-number*

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *mgmt-interface* **Example:** `Router(config)# interface gig1`	Enters interface configuration mode for the management interface.
Step 4	**ip address** *mgmt-ipv4-addr subnet-mask* **Example:** `Router(config-if)# ip address 172.25.29.235 255.255.255.128`	Configures the IP address for the management interface.
Step 5	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Enables the management interface.
Step 6	**exit** **Example:** `Router(config-if)# exit`	Exits interface configuration mode.
Step 7	**interface virtualportgroup** *virtual-port-group-number-number* **Example:** `Router(config)# interface virtuaportgroup 0`	Creates a virtual port group and enters virtual port group interface configuration mode.

	Command or Action	Purpose
Step 8	**ip unnumbered** *management-interface* **Example:** `Router(config-if)# ip unnumbered gigabitethernet1`	Enables IP processing on an interface without assigning it an explicit IP address.
Step 9	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Enables the management interface.
Step 10	**exit** **Example:** `Router(config-if)# exit`	Exits virtual port group interface mode.
Step 11	**virtual-service csr_mgmt** **Example:** `Router(config)# virtual-service csr_mgmt`	Configures the **csr_mgmt** virtual services container and enters virtual services configuration mode.
Step 12	**vnic gateway virtualportgroup** *virtual-port-group-number* **Example:** `Router(config-virt-serv)# vnic gateway virtualportgroup 0`	Creates a vNIC gateway interface for the virtual services container and maps the vNIC gateway interface to the virtual port group.
Step 13	**guest ip address** *remote-mgmt-ipv4-addr* **Example:** `Router(config-virt-serv-intf) guest ip address 172.25.29.236`	Configures the remote-management IP address for the vNIC gateway interface for the virtual services container.
Step 14	**exit** **Example:** `Router(config-virt-serv-intf)# exit`	Exits virtual services interface configuration mode and enters virtual services configuration mode.
Step 15	**activate** **Example:** `Router(config-virt-serv)# activate`	Activates the **csr_mgmt** virtual services container.
Step 16	**end** **Example:** `Router(config-virt-serv)# end`	Exits virtual services configuration mode and enters global configuration mode.

	Command or Action	Purpose
Step 17	**ip route** *ip-address subnet-mask* **virtualportgroup** *virtual-port-group-number* **Example:** `Router(config)# ip route 172.25.29.236 255.255.255.255 VirtualPortGroup0`	Creates an IP route that maps to the virtual port group. Use the same IP address that was configured using the **guest ip address** command.

Enabling Remote Management by the Cisco Prime Network Services Controller Host

Note The Cisco Prime Network Services Controller is unsupported using Cisco IOS XE Denali 16.3.1 or later, on the Cisco CSR 1000v.

The Cisco Prime Network Services Controller control point agent (CPA) is used to manage the interface between the Cisco CSR 1000v and the Cisco Prime Network Services Controller host. The Cisco Prime Network Services Controller CPA must be activated on the Cisco CSR 1000v before Cisco Prime Network Services Controller can be used to remotely manage the router.

You must use the Cisco IOS XE CLI to manually activate the Cisco Prime Network Services Controller CPA in the following situations:

- If you did not enable Cisco Prime Network Services Controller support through bootstrap when you deployed the OVA.
- If you are manually configuring the Cisco CSR 1000v when it is up and running.

For more information about installing the Cisco CSR 1000v by deploying the OVA, see Deploying the Cisco CSR 1000v OVA to the VM using vSphere, on page 79 and Deploying the Cisco CSR 1000v OVA to the VM using COT, on page 88.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. remote-management
4. **pnsc host** *ipv4-addr* **local-port** *number* **shared-secret** *string*
5. **end**
6. **show remote-management status**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	remote-management **Example:** `Router(config)# remote-management`	Enters remote-management configuration mode.
Step 4	**pnsc host** *ipv4-addr* **local-port** *number* **shared-secret** *string* **Example:** `Router(cfg-remote-mgmt)# pnsc host 172.25.29.234 local-port 8443 shared-secret ********`	Enables remote management by Cisco Prime Network Services Controller and sets up the access to the Cisco Prime Network Services Controller host. • The *ipvr-address* represents the IP address of the Cisco Prime Network Services Controller host. • The **local-port** is the TCP port number for receiving the HTTPS requests from Cisco Prime Network Services Controller. The valid range is from 1 to 65535. There is no default port number. The **local-port** number should not be the same port number configured with the **ip http port** command. • The **shared-secret** configured in this step should match the shared-secret configured on Cisco Prime Network Services Controller. Once configured, only the encrypted version of the shared secret is displayed. **Note** When remote management by Cisco Prime Network Services Controller is enabled using this command, the REST API PUT, POST, and DELETE operations are disabled. However, the GET operation is still available.
Step 5	**end** **Example:** `Router(config-remote-mgmt)# end`	Exits configuration mode and enters privileged EXEC mode.

	Command or Action	Purpose

Step 6	**show remote-management status**
	Example:
	`Router# show remote-management status`
	`RESTful-API: enabled`
	`https port: 443`
	`PNSC CPA: enabled`
	`Host 172.27.208125 port 8443 shared-secret *******`
		Displays the Cisco CSR 1000v remote management settings.

What to do next

Once remote management by Cisco Prime Network Services Controller is enabled, the following warning is displayed when entering the Cisco IOS XE CLI mode directly on the router:

```
WARNING: This device is managed by Prime Network Services Controller. RESTful API is read
only. Changing configuration using CLI is not recommended.
```

See documentation for Cisco Prime Network Services Controller.

Disabling Remote Management by the Cisco Prime Network Services Controller Host

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **remote-management**
4. **no pnsc host** *ipv4-addr* **local-port** *number* **shared-secret** *string*
5. **end**
6. **show remote-management status**

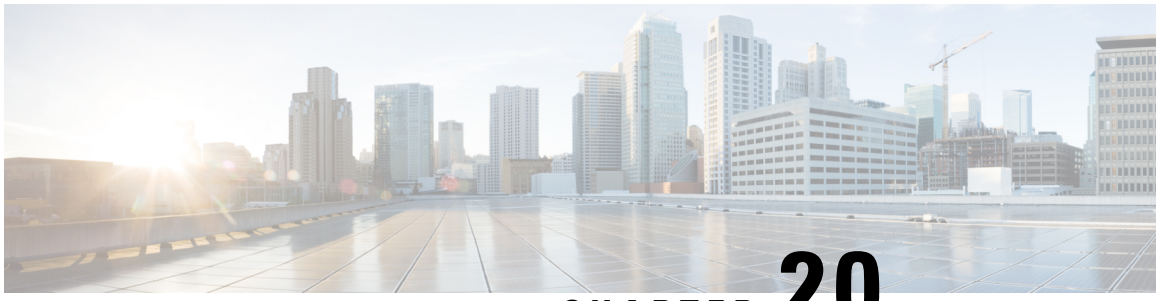
DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	`Router> enable`	
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**remote-management** **Example:** `Router(config)# remote-management`	Enters remote-management configuration mode.
Step 4	**no pnsc host** *ipv4-addr* **local-port** *number* **shared-secret** *string* **Example:** `Router(cfg-remote-mgmt)# no pnsc host 172.25.29.234 local-port 8443 shared-secret ********`	Disables remote management by Cisco Prime Network Services Controller. **Note** When remote management by Cisco Prime Network Services Controller is disabled using this command, the REST API PUT, POST and DELETE operations are enabled.
Step 5	**end** **Example:** `Router(cfg-remote-mgmt)# end`	Exits configuration mode and enters privileged EXEC mode.

Step 6	**show remote-management status**
	Example:
	`Router# show remote-management status`
	`RESTful-API: enabled`
	`https port: 443`
	`PNSC CPA: disabled`
	`Host 172.27.208.125 port 8443 shared-secret *******`
		Displays the Cisco CSR 1000v remote management settings.



CHAPTER 20

Performing a Factory Reset

This chapter provides information on performing a factory reset for a CSR 1000v instance. The factory reset feature helps remove any sensitive information from the router, or to reset the router to a fully functional state.

Information About Factory Reset

The factory reset is a process of clearing the current running and start up configuration information on a router, and resetting the router to an earlier, fully functional state. The factory reset process uses the **factory-reset all** command.

Note The time taken for factory reset on a CSR 1000v instance is dependent on factors such as the type of storage and the devices present on the router.

Information deleted:

When you perform a factory reset, the following information is deleted:

- Licenses – user installed, and manufacturer provided
- Non-volatile random-access memory data
- User credentials
- Start-up configuration
- All writable file systems and personal data
- ROMMON variable
- Persistent storage devices
- Any containers running on bootflash

Information retained:

However, the following information will be retained even after the factory reset:

- Critical information including files that provide access to the router after the reset is complete
- The software packages that are installed before you perform factory reset
- UDI and Smart Licensing files

Supported Scenarios:

You can use the factory reset feature in the following scenarios:

- When you want to delete a CSR 1000v instance in a secure manner.
- If the router data is compromised due to a malicious attack, you must reset the router to factory configuration and then reconfigure once again for further use.

Supported Platforms:

Factory reset is supported on a CSR 1000v instance running on all the platforms including Amazon Web Services, Microsoft Azure, GCP cloud, VMware ESXi, and Hyper-V.

Prerequisites for Performing Factory Reset

- Ensure that your CSR 1000v instance is running on 16.11.x release or later.
- Ensure that you take a backup of all the software images, configurations and personal data before performing the factory reset operation.
- Ensure that there is uninterrupted power supply when the feature reset process is in progress.
- Ensure that the instance has at least 8 GB memory in the bootflash.

Restrictions for Performing a Factory Reset

- Any software patches that are installed on the router are not restored after the factory reset operation.
- You must not restart the CSR 1000V instance during the factory reset process.
- If the factory reset command is issued through a Virtual Teletype (VTY) session, the session is not restored after the completion of the factory reset process.
- Any CSR 1000V instance running on 16.10.x release or earlier does not support this functionality.
- This feature is supported only when you launch the CSR 1000V instance created in the 16.9.x or the 16.10.x releases. If you launch the instance from any other earlier release and then upgrade to release 16.11 or above, factory reset is no longer supported.
- If you want to perform a factory reset for a CSR1000V instance that is running on Microsoft Azure, the instance should belong to a 16.11 release or later. If you upgrade from an earlier release, do not perform a factory reset. This feature is not supported in the earlier releases.

How to Perform a Factory Reset

Procedure

Step 1 Log in to a CSR 1000v instance running a 16.11 IOS-XE image or later.

Step 2 At the command prompt, execute the **factory-reset all** command.

The system displays the following:

```
factoryreset#factory-reset all
The factory reset operation is irreversible for all operations. Are you sure? [confirm]
 The following will be deleted as a part of factory reset:
 1: All writable file systems and personal data
 2: Licenses
 3: Configuration
 4: User Credentials
 The system will reload to perform a factory reset.
 Note that any day0 configuration will be applied after reload
 DO NOT STOP OR INTERRUPT THE POWER DURING RESET
Are you sure you want to continue? [confirm]Connection to 35.231.25.29 closed by remote host.
Connection to 135.231.25.29 closed.
```

Step 3 Enter confirm to proceed with the factory reset.

Note
The time taken for the factory reset process depends on the type of storage and on which cloud service you deploy the CSR instances.

Note
If you want to quit the factory reset process, press the **Escape** key.

What to do next

After the factory reset process is completed, you receive a log file in the bootflash that indicates whether the process was successful or not.

Restoring Smart Licensing after a Factory Reset

After the reset, Smart Licensing configuation is also deleted. You must reconfigure Smart Licensing on the router by using the token ID. In the connected mode, when you register your instance for Smart Licensing, you must use the force option. That is, you must use the **license smart register idtoken *****token***** force** command. The registration process begins.

When you do not use the force option, and configure Smart Licensing directly, the license registration fails. The following is an example of a failed registration output:

```
router#show license status
csr1#show license status
Smart Licensing is ENABLED

Utility:
```

```
  Status: DISABLED

Data Privacy:
  Sending Hostname: yes
    Callhome hostname privacy: DISABLED
    Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED

Transport:
  Type: Callhome

Registration:
  Status: UNREGISTERED - REGISTRATION FAILED
  Export-Controlled Functionality: NOT ALLOWED
  Initial Registration: FAILED on Feb 15 22:03:29 2019 UTC
    Failure reason: The product
regid.2013-08.com.cisco.CSR1000V,1.0_1562da96-9176-4f99-a6cb-14b4dd0fa135 and sudi containing
 udiSerialNumber:9XIVK9PIVPK,udiPid:CSR1000V has already been registered.

License Authorization:
  Status: No Licenses in Use

Export Authorization Key:
  Features Authorized:
```

After you execute the license smart register idtoken *****token***** force command, the license goes to the Registered state. The following is an example of a configuration output in the Registered state:

```
csr1#show license status
Smart Licensing is ENABLED

Utility:
  Status: DISABLED

Data Privacy:
  Sending Hostname: yes
    Callhome hostname privacy: DISABLED
    Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED

Transport:
  Type: Callhome

Registration:
  Status: REGISTERED
  Smart Account: InternalTestDemoAccount8.cisco.com
  Virtual Account: RTP-CSR-DT-Prod
  Export-Controlled Functionality: ALLOWED
  Initial Registration: SUCCEEDED on Feb 15 22:04:07 2019 UTC
  Last Renewal Attempt: None
  Next Renewal Attempt: Aug 14 22:04:06 2019 UTC
  Registration Expires: Feb 15 21:59:05 2020 UTC

License Authorization:
  Status: AUTHORIZED on Feb 15 22:04:11 2019 UTC
  Last Communication Attempt: SUCCEEDED on Feb 15 22:04:11 2019 UTC
  Next Communication Attempt: Mar 17 22:04:11 2019 UTC
  Communication Deadline: May 16 21:58:10 2019 UTC

Export Authorization Key:
  Features Authorized:
    <none>
```

What Happens after a Factory Reset

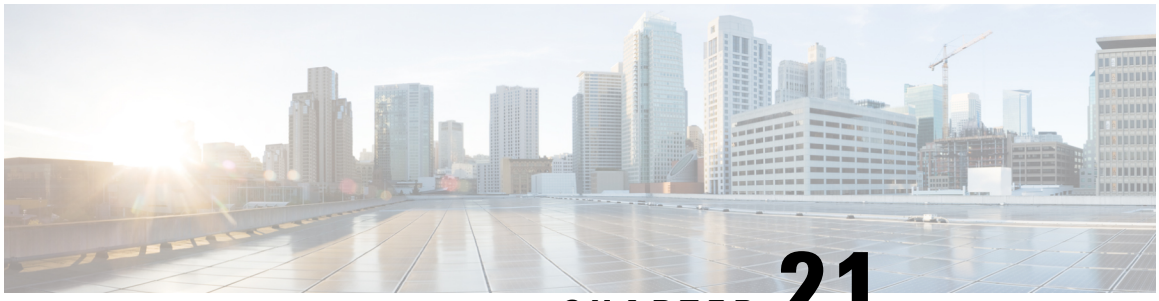
After the factory reset is successfully completed, the router boots up. However, before the factory reset process started, if the configuration register was set to manually boot from ROMMON, the router stops at ROMMON.

Important If the current boot image is a remote image or is stored in a USB or a NIM-SSD, ensure that you take a backup of the image before starting the factory reset process.

Factory reset does not change the UDI of the CSR 1000v instance. To verify whether the UDI is the same after the factory reset, execute the **factoryreset#show license udi** command before and after the factory reset process.

After you configure Smart Licensing, execute the **#show license status** command, to check whether Smart Licensing is enabled for your instance.

Note If you had SLR enabled before you performed the factory reset, use the same license and enter the same license key that you received from the smart agent.



CHAPTER 21

Configure High Availability

High Availability refers to the ability to establish redundancy of networking functionality and configuration data between two peer routers. This chapter provides overview information on high availability, and how you can configure high availability on a Cisco CSR 1000v instance running on different cloud service providers.

Overview of High Availability Version 3

Feature History Table

Release	Description
Cisco IOS XE Gibraltar 16.11.1	High availability version 3 introduced.
Cisco IOS XE Amsterdam 17.3.1	The `pip3 install csr-[aws/azure/gcp]-ha --user` script introduced to install high availability version 3.

The High Availability feature is supported for Cisco CSR 1000V Routers running on Microsoft Azure, Google Cloud Platform (GCP), and Amazon Web Services (AWS). A typical use case for the Cisco CSR 1000V is to interconnect two subnets within a virtual network. You can deploy Cisco CSR 1000V routers between the front-end (public) and the back-end (private) subnets. The Cisco CSR 1000V router represents a single point of failure for access to back-end resources. To mitigate this single point of failure, you must deploy two Cisco CSR 1000V routers between the two subnets.

The back-end subnet contains a routing table with entries pointing to the next hop router, which is one of the two Cisco CSR 1000V instances. The peer Cisco CSR 1000V routers communicate with one another over a tunnel using the Bi-directional Forwarding Detection (BFD) protocol. If the connection is lost between a

router and a peer, BFD generates an event. This event causes the active router that is working to update the entries in the route table so that the routing table points to the default route.

The routing table controls the upstream traffic of the Cisco CSR 1000V router and the routing protocol configured on the router determines the path of the downstream traffic.

In cloud environments, it is common for virtual networks to implement a simplistic mechanism for routing, which is based on a centralized route table. However, you can also create multiple route tables, where each route table has a subnet assigned. This subnet acts as the source of route information, and the route table is populated automatically which includes one or more individual routes depending on the network topology. You can also configure the routes in the route table.

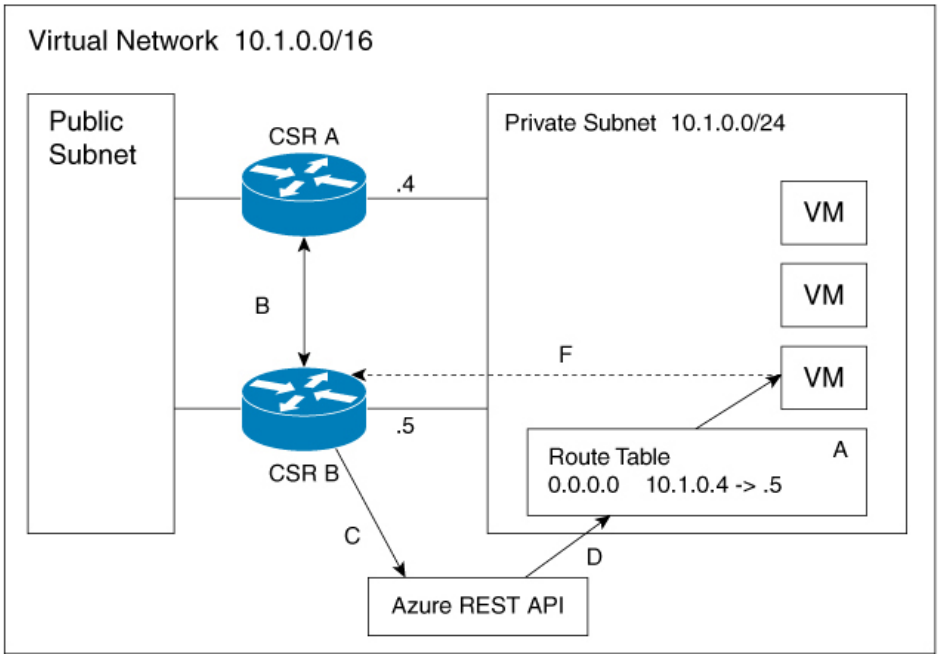
A subnet has a centralized route table, which allows two Cisco CSR 1000V routers to operate in a redundant mode. You can deploy two Cisco CSR 1000V routers in the same virtual network with their interfaces directly connected to subnets in the virtual network. You can add routes to the route table to point to one of the two redundant Cisco CSR 1000V routers. At any given time, one of the two Cisco CSR 1000V routers serves as the next-hop router for a subnet. This router is the active router for the subnet. The peer router is referred to as the passive router. The active router is the next hop for a given route destination.

The Cisco CSR 1000V router uses the Bi-directional Failure Detection (BFD) protocol to detect whether a peer router is operating properly. An IP tunnel is created between the two peer routers and each router periodically sends a BFD protocol message to the other router. If one router fails to receive a BFD message from the peer for a specific period, the active router concludes that the peer router has failed.

If the active router fails, the route table for the subnet can be dynamically updated to change the next hop address for one or more routes so that they refer to the passive router. If the peer router detects the failure of the active router, the peer router uses the programmatic API to update the route table entries.

For a route table entry, configure which of the two Cisco CSR 1000V routers is the “primary” router. The other router is the passive router if it is configured as a “secondary” router. By default, all routes are configured as secondary.

Figure 16: High Availability Version 3



The subnet on the right has an address block of 10.1.0.0/24. The two Cisco CSR 1000V routers that are connected to this subnet provide a redundant path for traffic leaving this leaf subnet. The subnet is associated with a route table which provides the route information to the virtual machines attached to the subnet.

Consider this scenario: Initially the default route in the route table has the IP address of the next hop router - 10.1.0.4 (CSR A). All the traffic leaving the subnet goes through CSR A. CSR A is currently the active router for the default route. When CSR A fails, CSR B detects the failure as this router stops receiving BFD protocol messages from CSR A. CSR B writes to the route table via a RESTAPI to change the default route to the interface of CSR B on the 10.1.0.0/24 subnet, which is IP address 10.1.0.5. CSR B then becomes the active router for the route to the 10.1.0.0 network.

Step	Description
A	CSR A with address 10.1.0.4 is the active router for the 10.1.0.0 network.
B	CSR A fails. CSR B detects the failure using the BFD protocol.
C	CSR B uses an HTTP request to the Azure REST API.
D	Azure updates the 10.1.0.0 route in the user-defined route table to the IP address of CSR B.
E	Virtual machines see the route table update.
F	Packets from the virtual machines are now directed to CSR B.

Topologies Supported

1-for-1 redundancy topology: If both the Cisco CSR 1000v routers have a direct connection to the same subnet, the routers provide a 1-for-1 redundancy. An example of 1-for-1 redundancy is shown in the preceding figure. All the traffic that is intended for a Cisco CSR 1000v only goes to one of the routers - the Cisco CSR 1000v that is currently active. The active Cisco CSR 1000v router is the next-hop router for a subnet. The other Cisco CSR 1000v router is the passive router for all the routes.

Load sharing topology: In this topology, both the Cisco CSR 1000v routers have direct connections to different subnets within the same virtual network. Traffic from subnet A goes to router A and traffic from subnet B goes to router B. Each of these subnets is bound to different route tables. If router A fails, the route table for subnet A is updated. Instead of router A being the next hop, the route entry is changed to router B as the next hop. If router B fails, the route table for subnet B is updated. Instead of router B being the next hop, the route entry is changed to router A as the next hop.

Redundancy Nodes

A redundancy node is a set of configuration parameters that specifies an entry in a route table. The next hop of a route is updated when an active router fails. To configure a redundancy node, you require the following information:

- Route Table – The identity of the route table in the cloud. Route table includes a region or group in which the table was created, an identifier for the creator or the owner of the table, and a name or identifier for

the specific table. Optionally, you can specify an individual route within the table. If you do not specify an individual route, the redundancy node represents all the routes in the table.

- Credentials - Authentication of the identity of the Cisco CSR 1000v router. Each cloud provider handles the process of obtaining and specifying the credentials differently.
- Next Hop - The next hop address that is written to the route entry when a trigger event occurs. Next Hop is usually the interface of the CSR 1000v routers on the subnet that is protected.
- Peer Router - Identifies the redundant router that will forward traffic for this route after a failure occurs on this router.
- Router Role—Identifies whether the redundancy node serves in a primary or secondary role. This is an optional parameter. If you do not specify this value, the router role defaults to a secondary role.

Event Types

The high availability feature recognizes and responds to three types of events:

- Peer Router Failure: When the peer route fails, it is detected as a Peer Router Failure event. In response to this event, the event handler writes the route entry with the next hop address that is defined in the redundancy node. To enable this event to be generated, configure the BFD protocol to a peer router and associate the BFD peer under redundancy for cloud high availability.
- Revert to Primary Router: After a router recovers from a failure, the *Revert to Primary Router* event occurs. The purpose of this event is to ensure that the primary router for the route is re-established as the active router. This event is triggered by a timer and you need not configure this event. In the route table entry, the event handler changes the next hop address that is defined in the redundancy node only if it is different from the next hop address that is currently set for the route.

  This Revert to Primary Router event is generated periodically using a CRON job in the guestshell environment. The job is scheduled to run every 5 minutes and checks if each redundancy node that is configured in the primary mode has this router's next hop interface set in the route table. If the route table entry already points to this router's next hop interface, then an update is not required. If a redundancy node configuration of the mode parameter is secondary, then the *Revert to Primary Router* event is ignored.
- Redundancy Node Verification: The event handler detects a Redundancy Node Verification event and reads the route entry that is specified by the redundancy node. The event handler writes the same data back to the route entry. This event is not generated automatically or algorithmically. This event verifies the ability of the event handler to execute its functions. Execute a script, manually or programmatically, to trigger the Redundancy Node verification event. For further information about the verification event, see *User-Defined Triggers*, in the *Advanced Programming for High Availability on Microsoft Azure* section.

High Availability Versions and OS Compatibility

Choose one of the following deployment options for High Availability on Cisco IOS XE Fuji 16.11.x and later:

- **High Availability Version 1**: This version is supported until Cisco IOS XE 16.11.1 release. From Cisco IOS XE 16.11.x, if you attempt to configure redundancy nodes in Cisco IOS, you receive a warning that the configuration is deprecated.

- **High Availability Version 2 with Redundancy Node Configuration in Cisco IOS XE**: You can configure High Availability Version 2 for Cisco CSR1000V 16.11.1 or later running on Microsoft Azure using CLI commands. This version provides access to the new features in HA version 2 and allows you to continue using your existing redundancy node configurations. However, this deployment option is deprecated from Cisco IOS XE 16.12.1.
- **High Availability Versions 2 and 3 with Redundancy Node Configuration in the guestshell**: You can configure high availability version 2 using guestshell-based Python scripts from Cisco IOS XE 16.9.1 release. The high availability version 3 is available from Cisco IOS XE 16.11.1 release, and this HA version is the recommended version.

  If you currently use redundancy node configuration in Cisco IOS XE, we recommend that you use the Redundancy Node configuration in the guestshell using the Cisco IOS XE Fuji 16.11.1 release.

Differences in High Availability across Cisco IOS XE Releases

The following table specifies the functionalities that are supported across the high availability versions.

Feature/Functionality	Suport in High Availability Version 1	Suport in High Availability Version 2	Suport in High Availability Version 3
Redundancy Node Configuration in IOS XE	Yes	Yes	Yes
Redundancy Node Configuration in Guestshell	No	Yes	Yes
Revert to the primary router after recovery	No	No	Yes
CSR1000V authentication by an application in Azure Active Directory	Yes	Yes	Yes
CSR1000V authentication by Managed Identity (formerly known as Managed Service Identity)	No	Yes	Yes

By default, in Cisco IOS XE Gibraltar 16.11.x, the Cisco CSR 1000v on AWS runs HA Version 2. To run HA Version 3, you must manually install and enable guestshell, and install the `csr_aws_ha` Python package in guestshell.

What's New in High Availability Version 3

The first version of high availability in the AWS cloud was introduced in Cisco IOS XE 16.3.1. The second version of high availability or HA Version 2 was released in Cisco IOS XE Fuji 16.5.1.

HA Version 3 is released in Cisco IOS XE Gibraltar 16.11.1. This high availability version supports several new features, a new configuration, and a deployment mechanism. Here's an overview of what's new in high availability version 3:

- **Cloud Agnostic**: This version of high availability is functional on CSR 1000v routers running on any cloud service provider. While there are some differences in the cloud terminology and parameters, the set of functions and scripts used to configure, control, and show the high availability features are common across the different cloud service providers. High Availability Version 3 (HAv3) is supported in CSR 1000v routers running on AWS, Azure, and GCP. Support for the GCP provider has been added in 16.11.1. Check with Cisco for current support of high availability in the individual provider's clouds.
- **Active/active operation**: You can configure both Cisco CSR 1000v routers to be active simultaneously, which allows for load sharing. In this mode of operation, each route in a route table has one of the two routers serve as the primary router and the other router as the secondary router. To enable load sharing, take all the routes and split them between the two Cisco CSR 1000v routers. Note that this functionality is new for AWS-based clouds.
- **Reversion to Primary CSR After Fault Recovery**: You can designate a Cisco CSR 1000v as the primary router for a given route. While this Cisco CSR 1000v is up and running, it is the next hop for the route. If this Cisco CSR 1000v fails, the peer Cisco CSR 1000v takes over as the next hop for the route, maintaining network connectivity. When the original router recovers from the failure, it reclaims ownership of the route and is the next hop router. This functionality is also new for the AWS-based clouds.
- **User-supplied Scripts**: The guestshell is a container in which you can deploy your own scripts. HAv3 exposes a programming interface to user-supplied scripts. This implies that you can now write scripts that can trigger both failover and reversion events. You can also develop your own algorithms and triggers to control which Cisco CSR 1000v provides the forwarding services for a given route. This functionality is new for AWS-based clouds.
- **New Configuration and Deployment Mechanism**: The implementation of HA has been moved out of the Cisco IOS XE code. High availability code now runs in the guestshell container. For further information on guestshell, see the "Guest Shell" section in the Programmability Configuration Guide. In HAv3, the configuration of redundancy nodes is performed in the guestshell using a set of Python scripts. This feature has now been introduced for AWS-based clouds.

Configure High Availability Version 3

The following sections specify the common configuration steps to configure High Availability Version 3 for a CSR 1000v running on any cloud service provider.

Configuring IOX and the Guestshell on Cisco IOS XE

The following Cisco IOS XE configuration shows the commands that are required to access the guestshell. You do not need to configure these prerequisites as they are included automatically in the startup-config file.

Procedure

Step 1 Perform the following configuration:

Example:

```
iox
   ip nat inside source list GS_NAT_ACL interface GigabitEthernet1 vrf GS overload
   ip route vrf GS 0.0.0.0 0.0.0.0 GigabitEthernet1 192.168.35.1 global
```

```
interface VirtualPortGroup0
  vrf forwarding GS
  ip address 192.168.35.101 255.255.255.0
  ip nat inside
  no mop enabled
  no mop sysid
ip access-list standard GS_NAT_ACL
  permit 192.168.35.0 0.0.0.255
app-hosting appid guestshell
  app-vnic gateway1 virtualportgroup 0 guest-interface 0
    guest-ipaddress 192.168.35.102 netmask 255.255.255.0
  app-default-gateway 192.168.35.101 guest-interface 0
  name-server0 8.8.8.8
```

Step 2 To configure High Availability, you must verify whether IOX is configured and running:

Example:

```
show iox
Virtual Service Global State and Virtualization Limits: Infrastructure version : 1.7
Total virtual services installed : 0 Total virtual services activated : 0 Machine types supported :
 LXC Machine types disabled : KVM Maximum VCPUs per virtual service : 1
Resource virtualization limits:
Name Quota Committed Available
------------------------------------------------------------
system CPU (%) 75 0 75
memory (MB) 3072 0 3072
bootflash (MB) 20000 0 5745
IOx Infrastructure Summary:
---------------------------
IOx service (CAF) : Running
IOx service (HA) : Not Running IOx service (IOxman) : Running Libvirtd : Running
```

Step 3 Enter the following command to verify that the guest application is defined and running:

Example:

```
show app-hosting list
App id State
------------------------------------------------------
guestshell RUNNING
```

If the state of the guestshell displays DEPLOYED in the output of the preceding command, you must enable the guestshell by using the following command:

```
guestshell enable
Interface will be selected if configured in app-hosting Please wait for completion
guestshell activated successfully Current state is: ACTIVATED guestshell started successfully Current
 state is: RUNNING Guestshell enabled successfully
```

Configure a Tunnel Between Cisco CSR 1000v Routers

You must configure a tunnel between the Cisco CSR 1000v routers and enable Bi-directional Forwarding Detection (BFD) and a routing protocol (EIGRP or BGP) on the tunnel for peer failure detection. To authenticate and encrypt IP traffic as it traverses a network, either use an IPsec tunnel or VxLAN GPE tunnel.

Procedure

Step 1 To configure an IPsec tunnel, enter the configuration mode commands to give the following configuration. The command crypto isakmp policy 1 defines an IKE policy, with a high priority (1), and enters config-isakmp configuration mode.

Example:

```
Crypto isakmp policy 1
encr aes 256 authentication pre-share
crypto isakmp key cisco address 0.0.0.0
!
crypto ipsec transform-set uni-perf esp-aes 256 esp-sha-hmac mode tunnel
!
crypto ipsec profile vti-1
set security-association lifetime kilobytes disable set security-association lifetime seconds 86400
  set transform-set uni-perf
set pfs group2
!
interface Tunnel1
ip address 192.168.101.1 255.255.255.252
load-interval 30
tunnel source GigabitEthernet1 tunnel mode ipsec ipv4
tunnel destination 23.96.91.169 tunnel protection ipsec profile vti-1
bfd interval 100 min_rx 100 multiplier 3
```

Step 2 To create a VxLAN GPE tunnel, enter the following configuration

```
interface Tunnel100
ip address 192.168.101.1 255.255.255.0
bfd interval 100 min_rx 100 multiplier 3 tunnel source GigabitEthernet1
tunnel mode vxlan-gpe ipv4 tunnel destination 40.114.93.164
tunnel vxlan vni 10000
```

For further information on configuring a VxLAN GPE tunnel, see: http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/cether/configuration/xe-16/ce-xe-16-book/vxlan-gpe-tunnel.html.

The tunnel destination address must be the public IP address of the corresponding Cisco CSR 1000v. For the tunnel IP address, use any unique IP address. However, the tunnel endpoints of each redundant Cisco CSR 1000v must be in the same subnet.

Note
To allow VxLAN to pass traffic through the tunnel, you must ensure that UDP ports 4789 and 4790 are allowed in the cloud's network security group. See the cloud provider's documentation for configuring network security filters.

Configuring EIGRP over Virtual Tunnel Interfaces

Configure EIGRP over the virtual tunnel interfaces using the following steps.

Note Other than using EIGRP, which is the protocol that is used in the following steps, you also have the option of using either BGP, or OSPF.

Before you begin

Configure either a VxLAN or IPsec tunnel between the Cisco CSR 1000v routers.

Procedure

Step 1 **router eigrp** *as-number*

Example:

```
Device(config)# router eigrp 1
```

Enables the EIGRP routing process and enters the router configuration mode.

Step 2 **network** *ip-address subnet-mask*

Share the network of the tunnel using EIGRP.

Example:

```
network 192.168.101.0 0.0.0.255
```

Step 3 **bfd all-interfaces**

Enables BFD globally on all the interfaces that are associated with the EIGRP routing process.

Example:

```
Device(config-router)# bfd all-interfaces
```

Step 4 **end**

Exits the router configuration mode and returns the router to the privileged EXEC mode.

Example:

```
Device(config-router)# end
```

Step 5 **show bfd neighbors**

Verifies that the BFD neighbor is active and displays the routing protocols that BFD has registered.

Example:

```
Device# show bfd neighbors

IPv4 Sessions
NeighAddr      LD/RD        RH/RS    State  Int
192.168.101.2  4097/4097    Up       Up     Tu100
```

Verify the Tunnel Surface

Procedure

To verify that the tunnel interface is configured and enabled, run the `show ip interface brief` command.

Example:

```
# show ip interface brief
                       IP-Address       OK? Protocol   Method   Status
GigabitEthernet1  192.168.35.20   YES DHCP  up up
GigabitEthernet2  192.168.36.12   YES DHCP  up up
Tunnel1           172.17.1.1      YES NVRAM up up
VirtualPortGroup0 192.168.35.101  YES DHCP  up up
```

Configure the BFD Peer Router

Procedure

Run the following command:

Example:

```
redundancy
cloud-ha bfd peer <peer_router_ip_address>
```

This configuration command identifies the peer router. The IP address is that of the peer Cisco CSR 1000v within the tunnel carrying the BFD protocol between the two CSR 1000v routers.

Install the High Availability Package

Procedure

Step 1 Execute the `#Router>guestshell` command to enter the guestshell.

Step 2 Install the appropriate Python package based on the cloud provider on which the CSR 1000v instance is running:

Cloud Provider	Package Name
Microsoft Azure	csr_azure_ha
Amazon Web Services	csr_aws_ha
Google Cloud Platform	csr_gcp_ha

Note
The package name for Microsoft Azure is the same for both HAv2 and HAv3. If you perform an install by executing the **pip install csr_azure_ha –-user** command, the latest HA V3 is downloaded.

Step 3 Install the package that is appropriate for your cloud service provider by using the **[guestshell@guestshell]$ pip install <package_name> --user** command.

Note

If you are a user who has newly installed Cisco CSR1000V Release 17.3.1, you must use **pip3 install csr-[aws]-ha --user** to install high availability version 3. If you are upgrading from 17.2 to 17.3.1, and you have already enabled the Guestshell, you need not use the latest python script.

Step 4 From the home directory, navigate to the subdirectory named `cloud:[guestshell@guestshell]$ cd cloud`.

Verify High Availability in Guestshell

Procedure

Execute the `[guestshell@guestshell cloud]$ systemctl status csr_ha` command.

Example:

```
[guestshell@guestshell cloud]$ systemctl status csr_ha
azure-ha.service - Azure High Availability service
Loaded: loaded (/etc/systemd/user/csr_ha.service; enabled; vendor preset: disabled) Active
active (running) since Wed 2018-06-13 19:56:00 UTC; 7min ago Main PID: 29 (python)
CGroup: /system.slice/libvirtd.service/system.slice/csr_ha.service
├─ 29 python
/home/guestshell/.local/lib/python2.7/site-packages/csr__ha/server/ha_serve...
└─103 python
/home/guestshell/.local/lib/python2.7/site-packages/csr_ha/server/ha_serve...
```

Note
If the service is not running, start the service by running the `[guestshell@guestshell azure]$ sudo systemctl start csr_ha` command.

Set the Path Environment Variable

Procedure

Step 1 Update the path environment in the guest shell to specify the location of the configuration scripts.

Step 2 Execute the `source ~/.bashrc` command in the guestshell.

Configure the Redundancy Nodes

You can use Python scripts to create and modify redundancy nodes. Python scripts use the parameters that are shown in the following table. These parameters are described here in general for edification. To see the actual parameters, see the “Configure Redundancy Nodes” section specific for each cloud provider section.

Name of the parameter	Is this parameter required?	Description
Node Index	Yes	The index that is used to uniquely identify this node. Valid values: 1–1023
Cloud Provider	Yes	Specifies the specific cloud offer from the cloud provider.
User Identifier	Yes	A cloud-specific parameter which identifies a user or subscriber to the cloud service.
Scope	Yes	Some cloud providers offer a mechanism to bundle resources in a logical container. This parameter is used to identify this container.
Route table	Yes	The name or identifier of the route table to be updated.
Route	No	IP address of the route to be updated in CIDR format. Can be IPv4 or IPv6 address. If a route is unspecified, then the redundancy node is considered to apply to all valid routes in the routing table. Different clouds may have restrictions on what constitutes a valid route.
Next hop address	Yes	Interface or IP address of the next hop router. Use the IP address that is assigned to this CSR 1000v on the subnet which utilizes this route table. Can be an interface with an IPv4 or IPv6 address.
Mode	No	Indicates whether this router is the primary or secondary router for servicing this route. Default value is secondary.

Show Node

Procedure

Run the following script to display the parameter values of an existing redundancy node: `show_node.py{ -i value }`.

Example:

```
show_node.py -i 10
```

Here, -i specifies the index of the redundancy node (1-1023).

If the script is successful, the system displays the parameters of the specified node. If the script is unsuccessful, the system displays an error message.

Delete a Node

Procedure

Execute the `delete_node.py{ -i value }` command to delete an existing redundancy node.

Example:

```
delete_node.py -i 10
```

Here, -i specifies the index of the redundancy node (0–255).

The -i parameter is mandatory. The node is expected to exist in the database. If the client tries to delete a node that is not in the database, an error is not generated. If successful, the script returns a value of zero and the system displays the parameters of the specified node. If the script is unsuccessful, a non-zero value is returned. An error message is written to the log file.

Note
The database of redundancy nodes is maintained on a virtual disk that is allocated to the guestshell. If you issue the guestshell destroy Cisco IOS XE command, the virtual disk is deleted and all the node configurations are lost.

Recover deleted virtual disk

Procedure

Step 1 Enable the guestshell.

Step 2 Install the appropriate HA Python package for your cloud provider.

Step 3 Run the Python scripts to reconfigure all the redundancy nodes. You can reapply the node configuration by using the `create_node` script.

Configuring High Availability in the Cloud Provider Network

To complete the configuration for the high availability service, you must also execute configuration changes in the cloud provider's network. This section discusses these configuration requirements in general. The details of these steps are specific to each cloud provider. For specific configuration steps, see the appropriate cloud provider sections configuration sections.

- **Authenticate the CSR 1000v**: For the CSR 1000 router to access any resource in the cloud, you must first provide the proof of its identity. That is, you must authenticate the router.
- **Grant Access to the Route Table**: You must authorize the router to read and write to the route table. This usually involves configuring some form of identity and access management on the route table itself.
- **Network Security**: Cloud providers usually support a mechanism for filtering traffic which can pass to and from various network resources such as subnets and network interfaces. To enable the exchange of the BFD protocol messages between the pair of peer routers, it is necessary for the network security group that is associated with the CSR interfaces hosting the tunnel to allow ports 4789 and 4790 to be passed.
- **Verify a Redundancy Node**: If all the above configuration is complete, you must verify that a route that is represented by a redundancy node is successfully updated by the CSR 1000v router. You can verify this by simulating a peer failure for this redundancy node. This is a diagnostic tool that verifies whether the CSR 1000v router is capable of reading and writing the route table that is specified by the redundancy node.

Trigger the failover Using EEM

You can configure your Cisco CSR1000V instance with an Embedded Event Manager (EEM) applet which can detect the transition of an interface state and trigger a failover of the Cisco CSR1000V instance.

To trigger the failover, enter configuration mode in the Cisco IOS XE CLI of your Cisco CSR 1000V instance. Run the following commands:

```
event manager applet Interface_GigabitEthernet2
event syslog pattern "Interface GigabitEthernet2, changed state to administratively down"
action 1 cli command "enable"
action 2 cli command "guestshell run node-event.py -i 10 -e peerFail" exit
exit
```

Note The above-mentioned EEM script is only a generic example. You must modify the script to suit your particular environment and to trigger on an event you specify. For example, you can modify the script to trigger on BFD messages or a line protocol state.

Set user-defined Triggers

You can write your own Python script to recognize an event or condition and call the node_event script. You can also enter the command manually at the guestshell prompt.

As a sample scenario, to process the redundancy node and update an associated route table entry, run the `node_event` Python script.

For example, `node_event.py -i node_index -e peerFail` processes the redundancy node, and updates the associated route table entry.

Configure High Availability for CSR 1000v Running on Azure

The first version of high availability in the Azure cloud was introduced in Cisco IOS XE Everest 16.5.1. The second version of high availability, or HA Version 2, was released in Cisco IOS XE Fuji 16.9.1. High

Availability Version 3 is supported on Cisco IOS XE Gibraltar 16.11.1 release and later. The configuration of the BFD peer router is simplified in HAv3.

Create Binding to BFD Peer

Procedure

When you configure High Availability Version2 with IOS XE releases 16.9.x to 16.11.x, you can create a binding to a BFD peer by executing the following command:

Example:

```
redundancy
cloud provider azure index
bfd peer <peerIpAddress>
```

When you configure either High Availability Version 2 or Version 3 with IOS XE releases 16.12.1 and later, you can create a binding to a BFD peer by executing the following command:

```
redundancy
cloud-ha bfd peer <peerIpAddress>
```

Configure Cloud Specific Redundancy Parameters

The following table specifies the redundancy parameters that are specific to Microsoft Azure:

Parameter Switch	Switch	Description
Node Index	-i	The index that is used to uniquely identify this node. Valid values: 1–255.
Cloud Provider	-p	Specifies the type of Azure cloud: azure, azusgov, or azchina.
Subscription ID	-s	The Azure subscription id.
Resource Group Name	-g	The name of the resource group that contains the route table.
Route Table Name	-t	The name of the route table to be updated.
Route	-r	IP address of the route to be updated in CIDR format. Can be IPv4 or IPv6 address. If a route is unspecified, then the redundancy node is considered to apply to all routes in the routing table of type “virtual appliance”.

Parameter Switch	Switch	Description
Next Hop Address	-n	The IP address of the next hop router. Use the IP address that is assigned to this CSR 1000v on the subnet which utilizes this route table. Can be an IPv4 or IPv6 address.
Mode	-m	Indicates whether this router is the primary or secondary router for servicing this route. Default value is secondary.

Create a Redundancy Node

Procedure

Run the following script to create a redundancy node and add it to the database: `create_node { switch value } […[{ switch value }].`

You must configure the following parameters for a valid redundancy node:

- Node Index
- Cloud Provider
- Subscription ID
- Resource Group Name
- Route Table Name

```
create_node -i 10 -p azure -s b0b1a9e2-4444-4ca5-acd9-bebd1e6873eb -g ds-rg -t ds-sub2-RouteTable -r
 15.0.0.0/8 -n 192.168.7.4
```

If the configuration is successful, the script returns a value of zero.

Set Redundancy Node Parameters

Procedure

To change the value of parameters in an existing redundancy node, run the following script: `set_params{ switch value } […[{ switch value }].`

Example:

```
set_params.py -i 10 -r 15.0.0.0/16 -n 192.168.7.5
```

The index parameter (-i) is mandatory. This command sets the values of the specified parameters. If the specified parameter is already defined for the redundancy node, the value of the parameter is updated.

```
set_params -i 10 -n 192.168.7.5 -m primary
```

In this example, the next hop address and mode will be updated for the redundancy node with index 10.

If this configuration is successful, the script returns a value of zero.

Clear Redundancy Node Parameters

Procedure

If you want to clear the value of specified parameters for an existing redundancy node, run the following script:

```
clear_params -i value { switch } […[{ switch }].
```

Example:

```
clear_params -i 10 -r -n
```

In this example, the clear_params script clears both the route and next hop address parameters.

Specify only the switch parameter when you clear an associated value. Do not include the current value of the parameter.

Note
Only the index parameter is required. The values of any additional specified parameters are cleared.

If the clearing is successful, the script returns a value of zero.

Authenticate the CSR 1000v

To update a routing table in the Azure network, you must first authenticate the CSR 1000v router. This is accomplished by creating an application which represents the CSR 1000v router in the Azure Active Directory. You can use the application that is granted permissions, to access the Azure network resources.

You can create the application by using the following two mechanisms:

- System-assigned managed identity - Azure automatically creates an application and binds it to the router. This mechanism was previously called as Managed Service Identity by Azure.
- Manual application registration in Azure Active Directory - Here, the user creates an application in the Azure Active Directory, which represents the CSR 1000v router.

You can manually create a managed identity in Azure Active Directory by creating an application which represents the router. The application is assigned a set of identifiers; tenant ID, application ID, and application key. These application identifiers must be configured in the high availability feature either as the default AAD application or within an individual redundancy node.

Alternatively, when you create the CSR 1000v, you can configure Azure to create a system-assigned managed identity for the CSR 1000v instance. In this case, you need not configure any application identifiers in the

high availability feature. That is, in the absence of the configuration of an application's tenant ID, application ID, and application key, the high availability feature assumes that the CSR 1000v router is using a system-assigned managed identity.

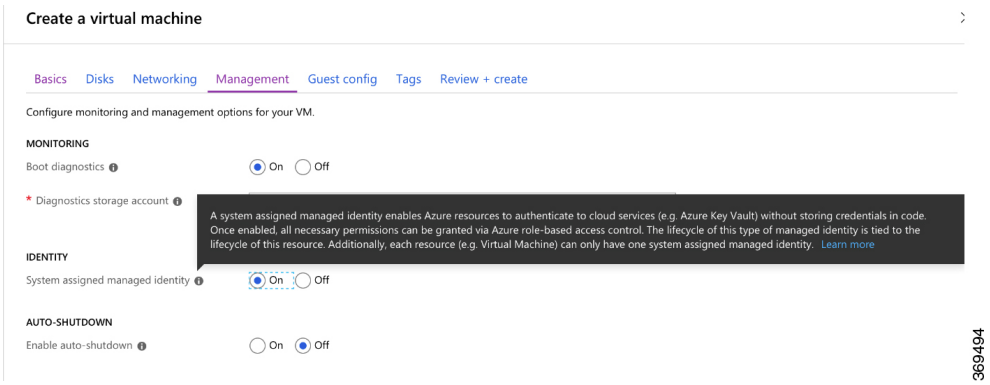
System Assigned Managed Identity

When you create the CSR 1000v router, you can enable for it to be assigned a system managed identity by Azure. There are two ways in which you can create a CSR 1000c router from the Azure marketplace:

- Solution template – A CSR 1000v router is created along with other Azure resources to create a networking solution in a single step.
- Standalone – A standalone CSR 1000v router is created, usually within an existing virtual network, with the base CSR image.

If you create a CSR 1000v router by using one of the solution template offerings in the Azure marketplace, a system-assigned managed identify for the CSR 1000v is enabled by default. If you create a standalone CSR 1000v by using a base CSR image, then a system-managed identity is enabled as shown in the following image:

Figure 17: Enable System Managed Identity



Authentication Using Azure Active Directory Service Principal

This section explains how to create an application in a Microsoft Azure Active Directory with permissions to access Microsoft Azure Resource Manager APIs.

Procedure

Step 1 See the latest instructions on registering an application with Azure Active Directory in Microsoft Azure documentation. See also: https://docs.microsoft.com/en-us/azure/active-directory/develop/quickstart-v1-add-azure-ad-app.

Step 2 Go to the portal for Microsoft Azure by visiting https://portal.azure.com.

Step 3 Choose your account name and sign in using your Microsoft Azure password.

Step 4 In the left navigation, click **Azure Active Directory** and select an Active Directory in the main pane. Click **Switch Directory** at the top of the pane to select the active directory.

Step 5 Verify whether you are authorized to create a new application. See the following Microsoft Azure documentation for creating an application in the Azure Active Directory: Use portal to create an Azure Active Directory application and service principal that can access resources.

Step 6 Navigate to the Active Directory that you want to use.

Step 7 To create a new application, select **Create** > **New Application Registration**.

Step 8 Specify the name of the application and ensure that **Web App / API** is selected as the Application type

Step 9 Specify the Sign-on URL. Use a name for the sign-on URL which is in the URI format, but it does not have to be reachable. You can use a string in the following format: `http://<your_directory_domain_name>/<app_name>`. For example, if your application name is myapp, and the domain name of your directory is `\mydir.onmicrosoft.com`, use the following is the sign-on URL: `http://mydir.onmicrosoft.com/myapp`.

Step 10 Click **Create**.

Step 11 Navigate to the Azure Active Directory page. Search for the application that you created. Make a note of the assigned Application ID.

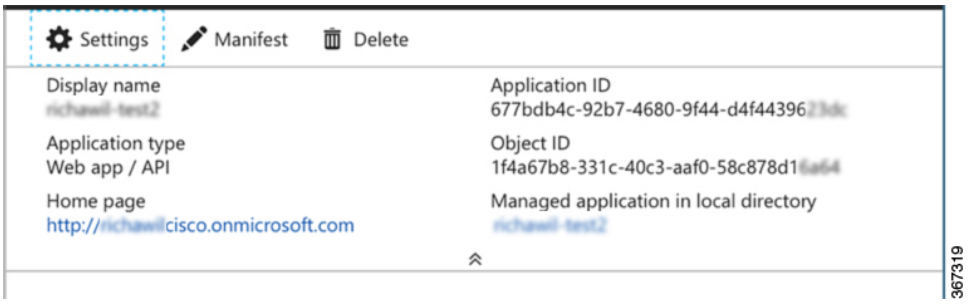
Obtain the Application ID and Tenant ID

Before you begin

Create an application in the Microsoft Azure Active Directory.

Procedure

Step 1 After you create the application, the registered app should appear on the screen as shown in the following image:



Step 2 Use the portal to create an Azure Active Directory application and service principal that can access resources. Make a note of the Application ID. See step 2 in the *Get application ID and authentication key* section in the Microsoft Documentation.

Step 3 Select **Azure Active Directory**.

Step 4 Select **Properties**. Make a note of the value in the **Directory ID** field. This is your tenant ID.

Create an Authentication key for the Application

Procedure

Step 1 From the Microsoft Azure portal, select the **Azure Active Directory**.

Step 2 Select **App Registrations**.

Step 3 Select the application that you previously created in the *Obtain the Application ID and Tenant ID* section.

Step 4 Click **Settings**.

Step 5 To create a key for API access, select **Keys** and specify a value for **Duration**. Duration is the length of time after which the key becomes invalid.

Step 6 Make a note of the API key from the **Value** field.

Caution
Store the API key carefully as it cannot be retrieved later.

Step 7 You must convert the API key to URL unencoded format. To find a suitable conversion tool, enter URL encoder into an Internet search engine. You might need the unencoded API key for procedures such as *Configure Failure Detection for the Cisco CSR 1000v on Microsoft Azure*.

Example:

```
URL encoded API Key:   5yOhH593dtD%2FO8gzAlWgulrkWz5dH02d2STk3LdbI4c%3D
         URL unencoded API Key:  5yOhH593dtD/O8gzAlWgulrkWz5dH02d2STk3LdbI4c=
```

Manage Azure Active Directory Applications in Guestshell

There are a set of utility scripts that can be run in the guestshell environment to manage applications in the Azure Active Directory, whether they were created manually as user-assigned identities or system-assigned identities. The following sections describe the use of these scripts and how to configure the binding between a redundancy node and the application used to authenticate the CSR 1000v router.

- Managing user-defined applications: If you have chosen to use a user-assigned identity for the CSR 1000v router, the application that was created in Azure Active Directory must be configured in the high availability feature. The application can be configured as the default application used for all the redundancy nodes, or for individual redundancy nodes.
- Set the default application: If you configure a user-assigned application as the default application using the `set_default_aad_app` script, all the redundancy nodes use the specified application for authentication, unless a redundancy node has an individual application configured.

Set the Default Application

Set the default application by running the `set_default_aad_app.py{ switch value } [...[{ switch value }]` script. See the following table for the AAD Redundancy Node Parameters:

Parameter Name	Switch	Description
Cloud Provider	-p	Specifies which Azure cloud is in use {azure | azusgov | azchina}
Tenant ID	-d	Identifies the AAD instance.
Application ID	-a	Identifies the application in AAD.
Application Key	-k	Access key that is created for the application. Key should be specified in unencoded URL format.

```
[guestshell@guestshell]$ set_default_aad_app.py -p azure -d
c4426c0b-036f-4bfb-b2d4-5c910c5389d6 -a 3d6e2ef4-8160-4092-911d-53c8f68ba808 -k
hZFvMGfzJuwFiukez27e/duyztom1bj7QL0Yix+KY9c=
```

```
[guestshell@guestshell]$ set_default_aad_app.py -h
usage: set_default_aad_app.py [-h] -p {azure,azusgov,azchina} -a A -d D -k K
AAD Application
required arguments:
  -p {azure,azusgov,azchina} <cloud_provider> {azure | azusgov | azchina}
  -a A                    to add the applicationId
  -d D                    to add the tenantId
  -k K                    to add the applicationKey
```

Clear the Default Application

You can clear the default user-assigned application configuration by using the `clear_default_aad_app` script.

```
[guestshell@guestshell]$ clear_default_aad_app.py
```

Clear the Application List

If you create a user-assigned application and associate the application with individual redundancy nodes, information about these applications is cached in memory. You can display the list of known applications by using the `show_auth_applications.py` script. Clear the cache using the clear_aad_application_list script.

```
[guestshell@guestshell]$ clear_aad_application_list.py
```

Managing all Applications

Use the following scripts to manage all the applications - user-assigned or system-assigned.

Showing Authentication Applications

The CSR 1000v router maintains a list of configured applications. You can view this list by using the show_auth_applications script.

```
[guestshell@guestshell]$ show_auth_applications.py
```

Clearing the Authentication Token

When an event is triggered on a redundancy node, the CSR 1000v router uses the configured application to obtain an authentication token from the Azure network. This token is cached up to five minutes in the router. You can clear the cached token by using the `clear_token` script.

This script clears either the default user-assigned application or the system-assigned application. The script does not clear the token on any user assigned application which is explicitly configured on an individual redundancy node.

```
[guestshell@guestshell]$ clear_token.py
```

Refreshing the Authentication Token

The CSR 1000v router can be forced to obtain a new token for the active application by using the `refresh_token` script.

This script refreshes either the default user-assigned application or the system-assigned application. This script does not refresh the token on any user-assigned application which is explicitly configured on an individual redundancy node.

```
[guestshell@guestshell]$ refresh_token.py
```

Select the Authentication Application

You can choose either system-assigned or user-assigned applications to identify a CSR 1000v router for the purpose of authentication. You can use the same mechanism for all the applications within a single CSR 1000v router. You can also have multiple user-assigned applications across multiple redundancy nodes.

The following table summarizes which application is used by the CSR 1000v router when processing a redundancy node:

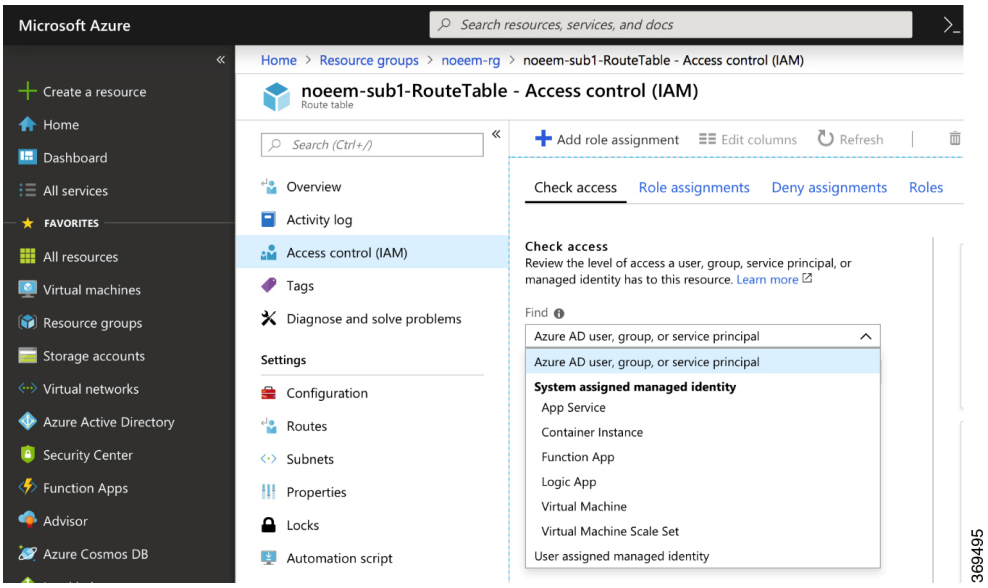
Table 52:

Is A Default Application Configured?	Does Node Have a User Assigned Application Configured?	Will CSR Use This Application?
No	No	System assigned application
No	Yes	User assigned application configured on this redundancy node
Yes	No	User assigned application configured as the default by set_default_aad_app.py
Yes	No	User assigned application configured on this redundancy node

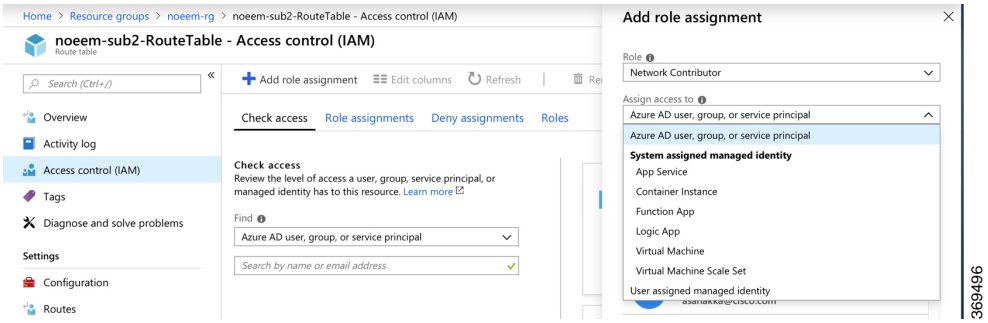
Configuring IAM for the Route Table

Procedure

Step 1 To add an application into an existing network, in the **All Resources** pane, choose a private side subnet from the left pane. For example, *noeem-sub1-RouteTable*.



Step 2 In the center pane, select **Access control (IAM)**. Select the plus icon to add a role assignment.



Step 3 In the Add Role Assignment screen, set the **Role to Network Contributor**.

Step 4 Select the **Assign Access to Pulldown** menu. If you are using system-assigned managed identity, select the **Virtual Machine** sub option and go to Step 6. If you are using user-assigned managed identity, select the option and go to step 5.

Step 5 In the **Select** field, enter the name of the user-assigned application that you created in **Azure Active Directory**. Click **Save**.

Step 6 In the **Select** field, enter the name given to the CSR 1000v instance. If you have configured the CSR 1000v instance properly for system-assigned identity, the CSR 1000v instance appears in the search results.

Step 7 Select the CSR 1000v instance by name, and click **Save**.

Route Table Entry Types

The route tables in Microsoft Azure support different entry types. The entry type for a route can be one of the following: Virtual network gateway, Internet, or Virtual Appliance. The next hop address identifies a resource in the Azure network.

Routes with an entry type of Virtual network gateway or Internet do not have an explicit IP address for the next hop and are not supported by the High Availability feature.

Cisco IOS XE Everest 16.6) When you configure High Availability on the Cisco CSR 1000v, all the routes within a route table must have an entry type of Virtual Appliance. These routes require an explicit IP address for the next hop.

(Cisco IOS XE Everest 16.7 or later) When you configure High Availability on the Cisco CSR 1000v, you can specify individual routes to be updated in the case of failure. Ensure that you configure each individual route as having an entry type of Virtual Appliance. If you configure a redundancy node that represents all the entries in the route table, ensure that all the routes have an entry type of Virtual Appliance.

Configuring the Network Security Group

If you have a network security group attached to NIC0 of the router, you must allow the BFD protocol to pass the interface. Configure an inbound and outbound security rule that allows ports 4789 and 4790 to be passed.

Configuring the Console Timeout

When you start an SSH session to the Cisco CSR 1000v router, ensure that you do not configure the terminal VTY timeout as infinite. That is, do not configure: exec-timeout 0 0. Use a non-zero value for the timeout; for example, exec-timeout 4 0. This command specifies a timeout of four minutes and zero seconds. The exec-timeout 0 0 command causes an issue as Azure enforces a timeout for the console idle period of 4 to 30 minutes. When the idle timer expires, Azure disconnects the SSH session. However, the session is not cleared from the point of view of the Cisco CSR 1000v as the timeout was set to infinite (by the exec-timeout 0 0 configuration command). The disconnection causes a terminal session to be orphaned. The session in the Cisco CSR 1000v remains open indefinitely. If you try to establish a new SSH session, a new virtual terminal session is used. If this pattern continues, the maximum number of simultaneous terminal sessions allowed is reached and no new sessions can be established. In addition to configuring the exec-timeout command correctly, it is also a good practice to delete idle virtual terminal sessions using the commands that are shown in the following example:

```
CSRA# show users
Line User Host(s) Idle Location
2 vty 0 cisco idle 00:07:40 128.107.241.177
* 3 vty 1 cisco idle 00:00:00 128.107.241.177
CSRA# clear line 2
```

Note If the workaround in the preceding scenarios are ineffective, as a last resort, you can restart the Cisco CSR 1000v router in the Azure portal.

Migrate from High Availability Version 1 to Version 3

In high availability version 3, configuration of redundancy nodes is no longer supported via the IOS configuration command line interface. Attempts to configure a redundancy node via IOS will result in a

warning message indicating such operations are deprecated. To migrate from high availability version 1 you must convert the IOS configuration to the format supported in guestshell.

If you are performing an in-place upgrade or a binary upgrade to the IOSXE release 16.11, then the configuration of redundancy nodes in IOS will be automatically copied to guestshell by performing the following steps:

If you are running the guestshell in the current IOS release, copy all important files out of the guestshell, as they will be erased as part of this upgrade.

Procedure

Step 1 In IOS executive mode, run the guestshell destroy command.

Step 2 Perform the binary upgrade to IOS XE release 16.11. Wait for the CSR 1000v router to reboot and stabilize.

Step 3 Enable the guestshell and enter the shell.

Step 4 Install the CSR HA package for Azure, version 3. For more information, see Installing the CSR HA Package.

Step 5 In IOS, configure the binding to the BFD peer router:

```
conf t
redundancy
cloud-ha bfd peer <peer-ip-addr>
end
```

This launches a process in the IOS to translate the configuration of redundancy nodes from IOS to guestshell. This process can take several minutes to complete, as each node is verified before the node is transferred.

Step 6 Wait for the transfer to complete. As each node is transferred, its configuration is added to a file in guestshell `~/cloud/HA/node_file`. When the size of this file stabilizes, the transfer is done. Open this file in the guestshell and examine its contents. Execute the script `cat node_file`.

Step 7 Verify whether all the nodes and their parameters have been successfully transferred. Use the create_node and the set_params scripts to make any adjustments.

Step 8 Restore any files copied out of the guestshell in step 1.

Step 9 Delete configuration of all redundancy nodes in IOS.

Migrate from High Availability Version 2 to Version 3

Procedure

Step 1 If you are currently running high availability version 2, a majority of the configuration of redundancy nodes should already be in guestshell. These nodes are stored in a file in the `~/azure/HA/node_file` location. Copy this file to bootflash by executing the `(guestshell#) cp ~/azure/HA/node_file /bootflash` command.

A redundancy node configuration in IOS still contains the binding between the node and the BFD peer. For example:

```
Router(config)#redundancy
Router(config-red)#cloud provider azure 4
Router(cloud-azure)#bfd peer 172.17.1.1
```

Step 2 To migrate this configuration to high availability version 3, add a global binding of all redundancy nodes to the BFD peer:

```
Router(config)#redundancy
Router(config-red)#cloud-ha bfd peer 172.17.1.1
And delete the individual redundancy nodes:
Router(config-red)#no cloud provider azure 4
```

Step 3 Delete all the existing nodes. This removes the nodes from the IOS configuration and deletes the same nodes configured in the guestshell. Once you delete all the nodes and you install the new HA version 3 python package, restore the configured nodes by copying the node file back by running the `(guestshell#) cp /bootflash/node_file ~/cloud/HA/node_file` script.

Step 4 Restart the High Availability setup after you copy the node_file. To manually restart the High Availability setup, run the `sudo systemctl start csr_ha` command.

Configure High Availability on CSR Running on Amazon Web Services

Table 53: Cloud Specific Configuration of Redundancy Parameters

Parameter	Switch	Description
Node Index	-i	Index that is used to uniquely identify this node. Valid values: 1–1023.
Region Name	-rg	Name of the region that contains the route table. For example, us-west-2.
Route Table Name	-t	Name of the route table to be updated. The name of the route table must begin with the substring rtb-. For example, rtb-001333c29ef2aec5f
Route	-r	If a route is unspecified, then the redundancy node is considered to apply to all routes in the routing table. The CSR cannot change routes which are of type local or gateway.

Parameter	Switch	Description
Next Hop Interface	-n	Name of the interface to which packets should be forwarded in order to reach the destination route. The name of the interface must begin with the substring eni-. For example, eni-07160c7e740ac8ef4.
Mode	-m	Indicates whether this router is the primary or secondary router for servicing this route. Valid values are primary or secondary. This is an optional parameter. The default value is secondary.

Create a Redundancy Node

Procedure

Run the following script to create a redundancy node and add it to the database.

Example:

```
create_node { switch  value } […[{ switch  value }]
```

A valid redundancy node must have the following parameters configured:

- Node Index
- Region Name
- Route Table Name
- Next Hop Interface Name

For example,

```
create_node.py -i 2 -t rtb-001333c29ef2aec5e -rg us-west-2 -n eni-07160c7e740ac8ef3 -r
2600:1f14:49b:9b03::/64
```

If successful, the script returns a value of zero.

Set Redundancy Node Parameters

Procedure

To change the value of parameters in an existing redundancy node, run the following script: `set_params -i node_index { switch value } […[{ switch value }].`

Example:

```
set_params.py -i 10 -r 15.0.0.0/16 -m primary
```

The index parameter (-i) is mandatory. This command sets the values of the specified parameters. If the specified parameter is already defined for the redundancy node, the value of the parameter is updated.

If this configuration is successful, the script returns a value of zero.

Clear Redundancy Node Parameters

Procedure

If you want to clear the value of specified parameters for an existing redundancy node, run the following script: `clear_params -i node_index {switch … switch].`

Example:

```
clear_params -i 10 -r -n
```

In this example, the clear_params script clears both the route and next hop address parameters.

Specify only the switch parameter when you clear an associated value. Do not provide the existing values for the parameters to be cleared.

If the clearing is successful, the script returns a value of zero.

Authenticate the CSR1000v Router

If you want the CSR 1000v router to update a routing table in the AWS network, you must first authenticate the router. In AWS, you must create a policy that permits the CSR 1000v router to access the route table. For example:

```
{
"Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogStream",
                "cloudwatch:",
```

```
                "s3:",
                "ec2:AssociateRouteTable",
                "ec2:CreateRoute",
                "ec2:CreateRouteTable",
                "ec2:DeleteRoute",
                "ec2:DeleteRouteTable",
                "ec2:DescribeRouteTables",
                "ec2:DescribeVpcs",
                "ec2:ReplaceRoute",
                "ec2:DescribeRegions",
                "ec2:DescribeNetworkInterfaces",
                "ec2:DisassociateRouteTable",
                "ec2:ReplaceRouteTableAssociation",
                "logs:CreateLogGroup",
                "logs:PutLogEvents"
            ],
            "Resource": "*"
        }
    ]
}
```

An IAM role is then created using this policy and applied to the EC2 resource.

After the CSR 1000v EC2 instances are created, the IAM role created above needs to be attached to each router.

Note See the AWS documentation for instructions on how to create policies, IAM roles, and how to associate a role to an EC2 instance.

Disable Source/Destination Address Checking

By default, network interfaces created in AWS have source and destination address checking enabled. The interface verifies all the traffic that passes through matches the source or destination address of the interface, otherwise it is dropped. For the CSR1000v to perform routing, this setting must be disabled on each CSR1000v interface.

Note See the AWS documentation for instructions on how to disable source/destination address checking on a network interface

Route Table Entry Types

The route tables in AWS cloud support different target types. These route targets include multiple types of gateways and connections. The CSR 1000v router is only capable of updating routes with a network interface target. Routes with other target types are ignored for the purposes of high availability.

If you configure a redundancy node without a specific route destination, the CSR 1000v attempts to update all the routes within a route table with a target type of network interface. All the other routes are ignored.

Configure Security Group

If you have a security group in use by the eth0 interface of the EC2 instance of the CSR 1000v, you must allow the BFD protocol to pass through the interface. Configure an inbound and outbound security rule that allows ports 4789 and 4790 to be passed.

Note See the AWS documentation for instructions on configuring security groups and attaching them to subnets and network interfaces.

Migrate from High Availability Version 2 to Version 3

In the high availability feature version 3, the configuration of redundancy nodes is temporarily supported via the IOS configuration command line interface. However, attempts to configure a redundancy node via IOS will result in a warning message indicating such operations are deprecated. To migrate from a CSR running high availability version 2 requires the configuration in IOS to be converted to the format supported in guestshell.

If you are performing an in-place upgrade or a binary upgrade to the IOS XE release 16.11.x, then the configuration of the redundancy nodes in IOS can be automatically copied to guestshell by performing the following steps:

If you are running the guestshell in the current IOS release, copy all important files out of the guestshell, as they will be erased as part of this upgrade. Files can be temporarily stored in /bootflash for example.

Procedure

Step 1 Perform the binary upgrade to IOS XE release 16.11. Wait for the CSR to reboot and stabilize.

Step 2 Enable the guestshell and enter the shell:

```
guestshell enable
guestshell
```

Step 3 Install the CSR HA package for AWS, version 3 by using the script (guestshell) pip install csr_aws_ha --user.

Note
If you are a user who has newly installed Cisco CSR1000V Release 17.3.1, you must use `pip3 install csr-[aws]-ha --user` to install high availability version 3. If you are upgrading from 17.2 to 17.3.1, and you have already enabled the Guestshell, you need not use the latest python script.

Step 4 In the IOS, configure the binding to the BFD peer router:

```
conf t
redundancy
cloud-ha bfd peer <peer-ip-addr>
end
```

This launches a process in the IOS to translate the configuration of redundancy nodes in IOS to guestshell. This process can take several minutes to complete, as each node is verified as it is transferred.

Step 5 Wait for the transfer to complete. As each node is transferred, its configuration is added to a file in guestshell `~/cloud/HA/node_file`. When the size of this file stabilizes, the transfer is done. Open this file in the guestshell and examine its contents by running the script `cat node_file`.

Step 6 Verify whether all the nodes and their parameters have been successfully transferred. Use the `create_node` and/or `set_params` scripts to make any adjustments.

Step 7 Restore any files copied out of the guestshell in step 1.

Step 8 Delete the configuration of all the redundancy nodes in IOS.

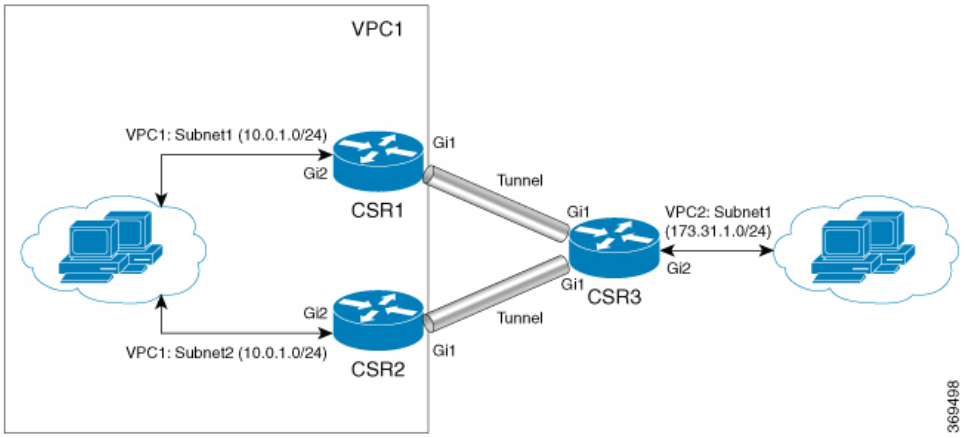
Configure High Availability in CSR 1000v Running On Google Cloud Platform

In the Google cloud, each static route belongs to the route table associated with a VPC and consists of following fields:

- Name and Description: These fields identify the route. A name is required, but a description is optional. Every route in your project must have a unique name.
- Network: Each route must be associated with exactly one VPC network.
- Destination range: The destination range is a single IPv4 CIDR block containing the IP addresses of systems that receive incoming packets. GCP does not support IPv6 destination ranges. Destinations must be expressed in CIDR notation, and the broadest destination possible is 0.0.0.0/0.
- Priority: Priority is used to determine which route should be used if multiple routes have identical destinations. Lower numbers indicate higher priorities; for example, a route with a priority value of 100 has a higher priority than one with a priority value of 200.
- Next hop: Static routes can have next hops that point to the default Internet gateway, a GCP instance, or a Cloud VPN tunnel. Refer to static route next hops for more information.
- Tags: You can specify a list of network tags so that the route will only apply to instances that have at least one of the listed tags. If you don't specify tags, GCP applies the route to all instances in the network.

For more information, see https://cloud.google.com/vpc/docs/routes. To configure High Availability in an active/active operation for two CSR 1000v routers in the Google network, you must create two routes in the route collection for each destination range, where each route points to one of the two routers as the next hop.

To understand this better, consider the following topology:



In the above topology, there are two routers configured in the HA mode. Both the routers have one interface in VPC1 and another in VPC. These two CSR 1000v routers have a Tunnel configured to another CSR that has an interface in VPC2. In this scenario, the following are the route entries in VPC1 for destination range of VPC 2 (172.31.0.0/16):

route-vcp2-csr1	172.31.0.0/16	100	None	IP:10:1:0:3	test-vpc
route-vcp2-csr2	172.31.0.0/16	200	None	IP:10.0.2.3	test-vpc

The active route is decided based on the route priority. Since route-vpc2-csr1 has a lower value, this route has a higher priority, thereby making CSR 1 as the active route.

Reversion to Primary CSR After Fault Recovery

If CSR 1 fails, CSR 2 detects a peer fail event through the BFD tunnel and deletes route-vpc2-csr1 from route collection making route-vpc2-csr2 as the active route for destination range 172.31.0.0/16.

When CSR 1 recovers, it adds route-vpc2-csr1 route back to the route collection which makes it the primary route again for all traffic to VPC 2. Please note it is possible to set equal route priority for both route entries in which case Google cloud uses both routes to send traffic to destination range.

On each CSR, you must create nodes corresponding to each route entry in route collection with next hop as the two CSRs.

When using mode (primary or secondary) option in HA to create a new node, ensure that the route with the higher priority (lower number) is marked as primary and the route with lower priority is marked as secondary.

User-Supplied Scripts

The guestshell is a container in which you can deploy your own scripts. HA Version 3 exposes a programming interface to user-supplied scripts, so you can write scripts that can trigger both failover and reversion events. You can develop your own algorithms and triggers to control which Cisco CSR 1000v provides the forwarding services for a given route.

Cloud Specific Configuration of Redundancy Parameters

Parameter	Is this parameter required?	Switch	Description
Node Index	Yes	-i	The index that is used to uniquely identify this node. Valid values: 1–255.
Cloud Provider	Yes	-p	Specify gcp for this parameter.
Project	Yes	-g	Specify the Google Project ID.
routeName	Yes	-a	The route name for which this CSR is next hop. For example from Fig. 2, if we are configuring node on CSR 1, this would be route-vpc2-csr1.
peerRouteName	Yes	-b	The route name for which the BFD peer CSR is next hop. For example from Fig. 2, if we are configuring node on CSR 1, this would be route-vpc2-csr2.
Route	yes	-r	The IP address of the route to be updated in CIDR format. Can be IPv4 or IPv6 address. If a route is unspecified, then the redundancy node is considered to apply to all routes in the routing table of type virtual appliance. Note: Currently Google cloud does not have IPv6 support in VPC.

Parameter	Is this parameter required?	Switch	Description
Next hop address	Yes	-n	The IP address of the next hop router. Use the IP address that is assigned to this CSR 1000v on the subnet which utilizes this route table. The value can be an IPv4 or IPv6 address. Note: Currently Google cloud does not have IPv6 support in VPC.
hopPriority	Yes	-o	The route priority for the route for which the current CSR is the next hop.
VPC	Yes	-v	The VPC network name where the route with the current CSR as the next hop exists.

Create a Redundancy Node

Procedure

Run the following script to create a redundancy node and add it to the database: `create_node { switch value } […[{ switch value }]`.

You must configure the following parameters for a valid redundancy node:

- Node Index
- Cloud Provider
- Project ID
- Route Name
- Peer Route Name
- Route
- Next Hop Address
- Hop Priority
- VPC Name

```
create_node -i 1 -g <project-id> -r dest_network -o 200 -n nexthop_ip_addr -a route-name1 -b route-name2
 -p gcp -v vpc_name
```

If the configuration is successful, the script returns a value of zero.

Set Redundancy Node Parameters

Procedure

To change the value of parameters in an existing redundancy node, run the following script: `set_params{ switch value } […[{ switch value }].`

Example:

```
set_params -i 10 -r 15.0.0.0/16 -n 172.168.7.5
```

The index parameter (-i) is mandatory. This command sets the values of the specified parameters. If the specified parameter is already defined for the redundancy node, the value of the parameter is updated.

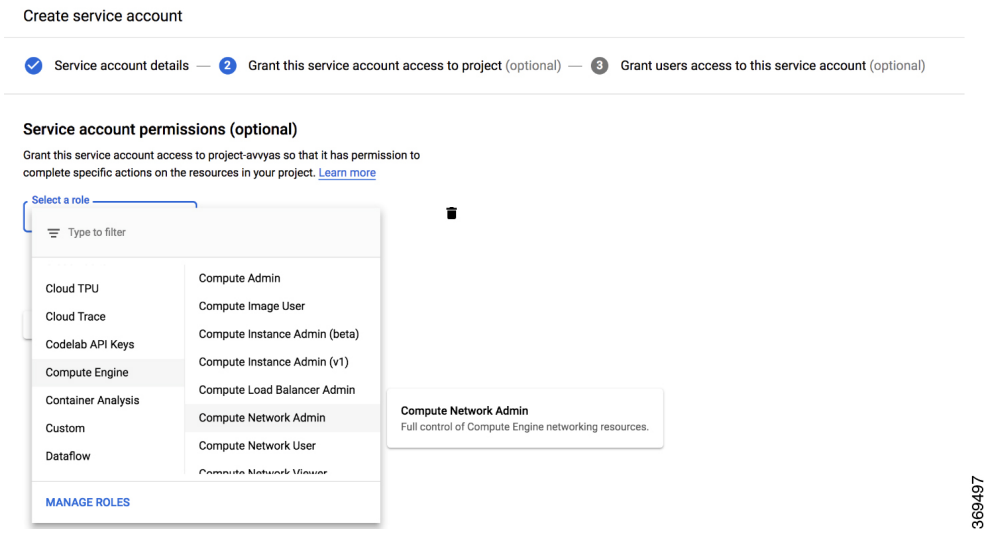
When a node index value of zero is specified, the values that are provided by the command for the specified parameters are treated as the default values for these parameters.

If this configuration is successful, the script returns a value of zero.

Authenticate the CSR 1000V Router

Procedure

Ensure that the service account associated with the CSR 1000v routers at least have a Compute Network Admin permission.



You can also provide the required permissions in a credentials file with name 'credentials.json' and place it under the /home/guestshell directory. The credentials file overrides the permissions supplied through the service account associated with the CSR 1000v instance.

Example Configurations

Example: Redunancy Nodes with Active/Active Configuration

Consider the HA configuration where route-name1 corresponds to route entry with next hop as CSR 1 and route-name2 corresponds to route entry with next hop as CSR 2 for destination network ‘dest_network’. To configure the routers in an active/active mode, set equal route priority for route-name1 and route-name2. In this case, Google cloud distributes the traffic between the routes using a five-tuple hash for affinity, thus implementing an ECMP routing design.

The node configuration on both routers corresponding to the route entries in Google route collection for the VPC would be:

```
create_node -i 1 -g <project-id> -r dest_network -o 200 -n nexthop_ip_addr_csr1 -a route-name1
 -b route-name2 -p gcp -v vpc_name

create_node -i 2 -g <project-id> -r dest_network -o 200 -n nexthop_ip_addr_csr2 -a route-name2
 -b route-name1 -p gcp -v vpc_name
```

Example: Redundancy Nodes with Active-Passive Configuration

Similarly, to configure CSRs in an active-passive mode, set the priority of one route higher than the other. In this case, Google cloud routes all the traffic from the VPC vpc_name to dest_network via the higher priority route (route-name1 for this example).

The node configuration on both routers corresponding to the route entries in Google route collection for the VPC would be:

```
create_node -i 1 -g <project-id> -r dest_network -o 200 -n nexthop_ip_addr_csr1 -a route-name1
 -b route-name2 -p gcp -v vpc_name
```

```
create_node -i 2 -g <project-id> -r dest_network -o 400 -n nexthop_ip_addr_csr2 -a route-name2
 -b route-name1 -p gcp -v vpc_name
```

Verify High Availability

Perform the following verification procedure by checking the log files. You can write a verbose log file to the directory ~/cloud/HA/events. Examine this log file to verify whether the operation is successful.

```
[guestshell@guestshell events]$ node_event.py -i node_index -e verify
[guestshell@guestshell events]$ cd /home/guestshell/cloud/HA/events
[guestshell@guestshell events]$ ls event.2018-06-13 20:10:21.093942
```

Troubleshoot High Availability Issues

Open the event file that is generated. This file is a debug log of the attempt to read and update the route described by the redundancy node. If the HA setup works as expected, the configuration output displays the status *Event handling completed*. If the system does not display this status, examine the log file in detail to determine which step of the verification failed.

Some of the common causes for failure include:

- Inability to obtain authentication credentials.
- The guestshell does not have network access.
- The authentication service is not running in Guestshell.
- The credentials for the CSR 1000v router are missing or incorrect.
- The router cannot access the route table entry.
- The route table was not correctly identified in the redundancy node
- The router was not granted permission to access the route table
- The specific route specified in the redundancy node does not exist

Note Cisco recommends that you use the `node_event` script with the verify event to test the configuration and the operation of the redundancy node.

Example: Troubleshooting Issues for High Availability Version 3

Execute the following command: `router#show iox`. See the following examples that provide the possible issues and how you can check and resolve these issues:

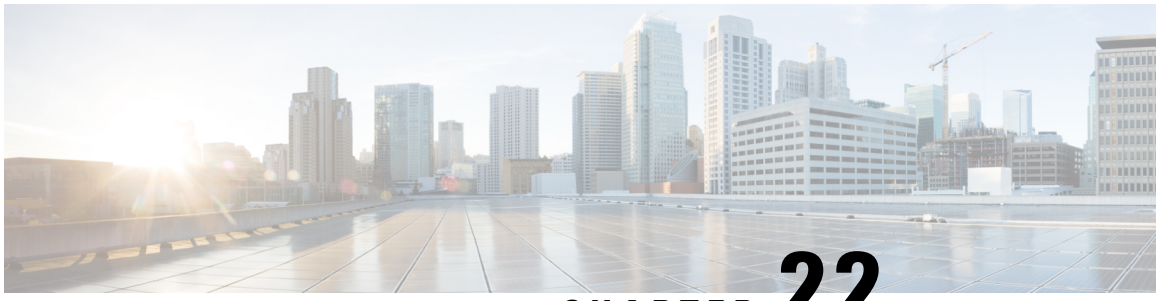
```
CSR#show iox

IOx Infrastructure Summary:
---------------------------
IOx service (CAF)    : Running
IOx service (HA)     : Not Supported
IOx service (IOxman) : Running
```

```
Libvirtd             : Running

CSR#guestshell enable

CSR#show app-hosting list
App id                           State
---------------------------------------------------------
guestshell                       RUNNING

CSR#guestshell
[guestshell@guestshell ~]$

[guestshell@guestshell ~]$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=38 time=25.7 ms

Possible Cause:
The configuration of IOX and the creation of the VirtualPortGroup interface to provide the
 guestshell network access is part of the "day zero" configuration of the CSR.  If any of
the above steps did not work, check that the startup configuration of the CSR has been
altered.

How to Fix:
A reload of the CSR will re-apply the day zero configuration.

---------

Problem:
HA package installation failure

How to Check:
CSR#guestshell
gsday0-csr#guestshell
[guestshell@guestshell ~]$ ls
cloud
[guestshell@guestshell ~]$ cd cloud
[guestshell@guestshell cloud]$ ls
HA

You should see the directory ~/cloud/HA.
On an Azure provided cloud, you should also see a ~/cloud/authMgr directory.

Possible Cause:
The HA package was not installed, or was not installed using the --user option.

How to Fix:
Install the package and set up the environment:
 pip install csr_<provider>_ha --user
 source ~/.bashrc

---------

Problem:
HA server not running.

How to Check:
[guestshell@guestshell ~]$ systemctl status csr_ha
● csr_ha.service - CSR High Availability service
   Loaded: loaded (/etc/systemd/user/csr_ha.service; enabled; vendor preset: disabled)
   Active: active (running) since Mon 2019-04-08 15:01:51 UTC; 2h 1min ago
 Main PID: 286 (python)
   CGroup: /system.slice/libvirtd.service/system.slice/csr_ha.service
           ├─286 python /home/guestshell/.local/lib/python2.7/site-packages/c...
           └─295 python /home/guestshell/.local/lib/python2.7/site-packages/c...
```

```
On an Azure provided network, the auth-token service should also be running.
[guestshell@guestshell ~]$ systemctl status csr_ha
● csr_ha.service - CSR High Availability service
   Loaded: loaded (/etc/systemd/user/csr_ha.service; enabled; vendor preset: disabled)
   Active: active (running) since Mon 2019-04-08 15:01:51 UTC; 2h 1min ago
 Main PID: 286 (python)
   CGroup: /system.slice/libvirtd.service/system.slice/csr_ha.service
           ├─286 python /home/guestshell/.local/lib/python2.7/site-packages/c...
           └─295 python /home/guestshell/.local/lib/python2.7/site-packages/c...
[guestshell@guestshell ~]$ systemctl status auth-token
● auth-token.service - Authentication Token service
   Loaded: loaded (/etc/systemd/user/auth-token.service; enabled; vendor preset: disabled)
   Active: active (running) since Mon 2019-04-08 16:08:15 UTC; 57min ago
 Main PID: 542 (python)
   CGroup: /system.slice/libvirtd.service/system.slice/auth-token.service
           └─542 /usr/bin/python /home/guestshell/.local/lib/python2.7/site-p...

Possible Cause:
If the HA server has an error and crashes, it is automatically restarted.

How to Fix:
A service can be restarted manually
[guestshell@guestshell ~]$ sudo systemctl start csr_ha


---------

Problem:
CSR authentication not working on Azure.
This is an Azure specific error.

How to check:
If you perform a node_event on a redundancy node, and it fails while trying to read the
route table, it will generate a file ~/cloud/HA/events/routeTableGetRsp.
[guestshell@guestshell ~]$ cat routeTableGetRsp
{"error":{"code":"AuthenticationFailedMissingToken","message":"Authentication failed. The
'Authorization' header is missing the access token."}}

Possible Cause:
There are multiple possible causes.  And it depends upon the authentication mechanism you
are using:
 - System assigned managed identity
 - Registered application in Azure Active Directory (AAD)

Likely cause of a failure using system assigned managed identity is that it is not enabled
 on the CSR.

How to Fix:
Verify the CSR is enabled for system assigned managed identity.
In the Azure portal, navigate to the virtual machine running the CSR.
Under the Settings menu, select the Identity item.
Under the system assigned tab, verify the status is set to On.

When using AAD for authentication, the likely cause of the error is a mis-configuration of
 the application or a mis-match in the identifiers for the application configured in the
guestshell.

How to Fix:
The application in AAD must be given the proper permissions to read and write a route table.
In the Azure portal, navigate to the registered application you have created.
Under the API Access menu, select the Required permissions item.
Select the Windows Azure Active Directory API.  In the Enable Access pane, verify the
following permissions are set:
```

```
 - Application permission to read and write directory data
 - Delegated permission to sign in and read user profile
Select the Windows Azure Service Management API.  In the Enable Access pane, verify the
following permissions are set:
 - Delegated permission to access Azure service management as organization users

How to Fix:
In the Azure portal, navigate to the registered application you have created.
Select the Setting button for the application.
Verify the application_id, tenant_id, and application key in the portal match the values
configured in guestshell.  Verify the application key configured in guestshell is in URL
unencoded format.

-------------

Problem:
Route table entry not updated by a peer failure event.

How to Check:
For every node event a log file is generated in the directory ~/cloud/HA/events.
This file will indicate the event that was processed and its result.  Examine this file for
 possible errors.  It is likely in the case of an error that a file
~/cloud/HA/events/routeTableGetRsp is also written.  Also examine this file for additional
 insights.

Possible Causes:
A route was not correctly identified in a redundancy node.  Depending upon what parameter
in the redundancy node is in error, you may see different results.

Some examples:
[guestshell@guestshell events]$ cat routeTableGetRsp
{"error":{"code":"SubscriptionNotFound","message":"The subscription
'b0b1a9e2-444c-4ca5-acd9-bebd1e6874ef' could not be found."}}
This implies the Azure subscription ID was not entered correctly.

[guestshell@guestshell events]$ cat node*
Route GET request failed with code 403
Route table get response:
 {"error":{"code":"AuthorizationFailed","message":"The client
'b3ce41c0-bcef-41d7-9741-26bea31221c1' with object id 'b3ce41c0-bcef-41d7-9741-26bea31221c1'
 does not have authorization to perform action 'Microsoft.Network/routeTables/read' over
scope
'/subscriptions/b0b1a9e2-444c-4ca5-acd9-bebd1e6873eb/resourceGroups/gsday0-rg/providers/Microsoft.Network/routeTables/gsday0-sub4-RouteTable'."}}
Route table not found.
This implies the name of the route table was incorrect or does not exist.

[guestshell@guestshell events]$ cat node*
Did not find route 17.0.0.0/8 event type peerFail
This implies that the route does not exist.

How to Fix:
Make sure the identifiers in the redundancy node match the values in the cloud provider's
portal.

------------

Problem:
Route table entry not updated by a peer failure event.

How to Check:
For every node event a log file is generated in the directory ~/cloud/HA/events.
This file will indicate the event that was processed and its result.  Examine this file for
 possible errors.  It is likely in the case of an error that a file
~/cloud/HA/events/routeTableGetRsp is also written.  Also examine this file for additional
```

```
 insights.

Possible Causes:
The CSR has not been given permission to access the route table.
Fetching the route table
Route table get response:
 {"error":{"code":"AuthorizationFailed","message":"The client
'b3ce41c0-bcef-41d7-9741-26bea31221c1' with object id 'b3ce41c0-bcef-41d7-9741-26bea31221c1'
 does not have authorization to perform action 'Microsoft.Network/routeTables/read' over
scope
'/subscriptions/b0b1a9e2-444c-4ca5-acd9-bebd1e6873eb/resourceGroups/gsday0-rg/providers/Microsoft.Network/routeTables/gsday0-sub2-RouteTable'."}}
Route GET request failed with code 403
Route table get response:
 {"error":{"code":"AuthorizationFailed","message":"The client
'b3ce41c0-bcef-41d7-9741-26bea31221c1' with object id 'b3ce41c0-bcef-41d7-9741-26bea31221c1'
 does not have authorization to perform action 'Microsoft.Network/routeTables/read' over
scope
'/subscriptions/b0b1a9e2-444c-4ca5-acd9-bebd1e6873eb/resourceGroups/gsday0-rg/providers/Microsoft.Network/routeTables/gsday0-sub2-RouteTable'."}}
Route table not found.
CSR HA: Set route table for verify
Route Table not found


If none of these troubleshooting tips have resolved your problem, run this command:
[guestshell@guestshell ~]$ cd ~/cloud/HA
[guestshell@guestshell ~]$ bash debug_ha.sh
[guestshell@guestshell ~]$ ls /bootflash
You should see a file name ha_debug.tar.  Copy this file off the CSR and provide it to Cisco
 Technical Support for analysis.
```



CHAPTER 22

Configuring VRF Route Sharing

The following chapter describes how you can configure VRF Route Sharing on a CSR 1000v instance. VRF Route Sharing is required when you need to forward traffic between an On-Premise Site and a Public Cloud Site. Configure VRF Route Sharing across VxLAN peers to deploy shared services across the cloud.

Information About VRF Route Sharing

In a hybrid cloud solution where there is an APIC layer (On-Premise) and a Public Cloud Site, the CSR 1000v instance connects the Data Centers through Layer-3 boundaries. The CSR 1000v instance has a VRF instance configured with two sets of import and export route-targets. One set of the import/export route target is associated with the BGP EVPN session with VXLAN encapsulation and L3 routing information in the On-Premise router. The other set of import/export route-target is associated with the L3VPN BGP neighbour in the service provider network. The CSR 1000v instance enables the L3 traffic movement across the EVPN by stitching the route between the On-Premise site and the service provider network.

The CSR 1000v instance forwards traffic across the EVPN even if the VRFs have the same VTEP IP (VxLAN tunnel endpoint) and RMAC (router MAC address). With this feature, the CSR1000v instance uses a binding label to setup the routing and forwaring chain.

Using the VRF Route Sharing functionality, you can deploy shared services across hybrid clouds. The shared services that run on the public cloud can be consumed by the endpoints on the On-Premise Site. The CSR 1000v instance shares the L3 prefix to multiple VRFs on the On-Premise Site, and vice versa. The APIC layer imports the addresses and the services are thus consumed in the APIC side.

Platforms Supported

The VRF Route Sharing functionality is currently supported only on Cisco CSR 1000v.

Limitations of VRF Route Sharing

- The VRF Sharing functionality supports up to 32 common VRFs, and 1000 customer VRF combination.

- This functionality does not support RT filters.
- VRF Route Sharing is supported only for IPv4 addresses and not IPv6 addresses.

Prerequisites of VRF Route Sharing

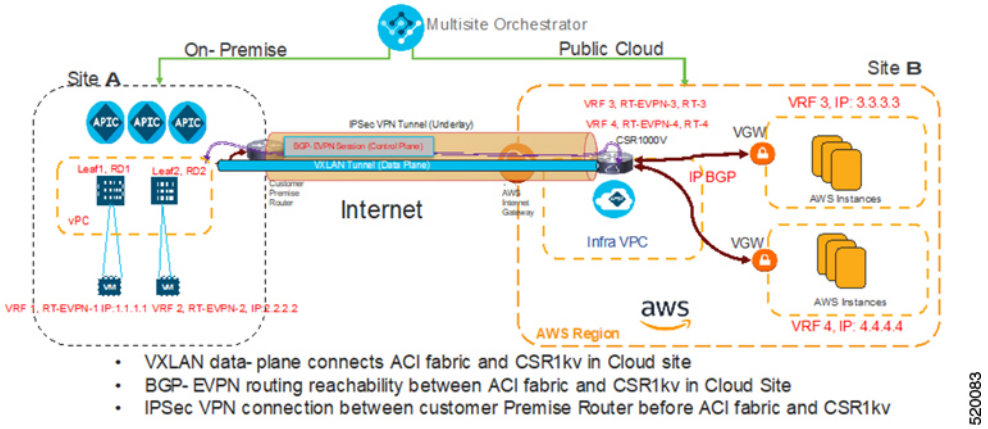
Before you configure the VRF Route Sharing functionality to enable the traffic between the ACI and the public cloud, ensure that:

- You configure VRF1 and VRF2 on the vPC pair of ACI.
- VRF3 are VRF4 on the CSR 1000v instance which peers with VGW have two RTs for each VRF.
- The CSR 1000v instance imports EVPN routes of VRF1&2 from ACI into VRF3&4.
- The IP BGP on the CSR 1000v side redistributes the routes to the gateway in the public cloud.
- The next-hop of routes from ACI are the spine of the border leaf of the ACI.
- There are no overlaps of prefix across the Route Sharing VRF.
- Advertise the L3 VPN routing and to forward the VRF prefixes to the EVPN neighbours. Run the advertise l2vpn evpn command and and export stitching RTs to push the native routes towards the EVPN.

How to Configure VRF Route Sharing

Sample Topology and Use Cases

The following is a sample topology that is used as an example to explain the VRF Route Sharing in a hybrid cloud:



In the above-mentioned topology, the CSR 1000v instance is deployed on the VM of the public cloud. Site A is an ACI deployment site, while Site B is the public cloud. Leaf 1 and Leaf 2 are the Virtual Port Channel (vPC) pair for ACI. These two vPCs are configured with different Route Distinguishers (RD). Here, VRF 1 and VRF 2 are configured on the vPC pair for ACI. For example,

VRF1 - RT:RT-EVPN-1, prefix:1.1.1.1

VRF2 - RT:RT-EVPN-2, prefix:2.2.2.2

VRF3 and VRF4 are configured on the CSR 1000v instance. These two VRFs pair with the Voice Gateway (VGW), and these two VRFs have two different Route Targets (RT). For example,

VRF3 – RT for EVPN: RT-EVPN-3, RT for IP BGP: RT-3, prefix:3.3.3.3

VRF4 – RT for EVPN: RT-EVPN-4, RT for IP BGP: RT-4, prefix:4.4.4.4

In the topology, the BGP-EVPN fabric is present between the ACI and the CSR 1000v instance in the public cloud and the IP BGP protocol is used between the CSR 1000v instance and the Cloud Service Provider such as Azure. The BGP-EVPN fabric redistributes the stitching routes between the EVPN and the IP BGP.

To enable the traffic flow between the ACI Site and the Public Cloud, both ACI and the CSR 1000v instance need to support VRF Route Sharing.

The CSR 1000v instance must be able to import the EVPN routes of VRF1 and VRF2 from ACI into VRF3 and VRF4. The IP BGP on the CSR side then redistributes the routes to the VGW in the public cloud.

Note When the VTEP (VxLAN Tunnel Endpoint) IP and the RMAC (Route MAC addrress) are the same for two leafs, and the VNIC alone differs, the CSR1000v instance can forward the traffic across the tunnel.

Use Cases

Using the same sample topology, here are the use cases for configuring VRF Route Sharing in a CSR 1000v instance:

- When VRF1 and VRF2 can talk to VRF3, but VRF3 and VRF4 cannot talk to each other, perform the following configuration:

```
vrf definition VRF3
rd 300:1
address-family ipv4
route-target export RT-EVPN-3 stitching
route-target import RT-EVPN-1 stitching
route-target import RT-EVPN-2 stitching
vrf definition VRF4
rd 400:1
address-family ipv4
```

- When VRF1 and VRF2 can talk to VRF3&4, but VRF3 and VRF4 cannot talk to each other, perform the following configuration:

```
vrf definition VRF3
rd 300:1
address-family ipv4
route-target export RT-EVPN-3 stitching
route-target import RT-EVPN-1 stitching
route-target import RT-EVPN-2 stitching
vrf definition VRF4
rd 400:1
address-family ipv4
route-target export RT-EVPN-4 stitching
route-target import RT-EVPN-1 stitching
route-target import RT-EVPN-2 stitching
```

- When VRF1 and VRF2 can talk to VRF3, but VRF3 and VRF4 can talk to each other, perform the following configuration:

```
vrf definition VRF3
rd 300:1
address-family ipv4
route-target export RT-EVPN-3 stitching
route-target import RT-EVPN-1 stitching
route-target import RT-EVPN-2 stitching
route-target export RT-3
route-target import RT-4
vrf definition VRF4
rd 400:1
address-family ipv4
route-target import RT-3
route-target export RT-4
```

- When VRF1 and VRF2 can talk to VRF3&4, but VRF3 and VRF4 can talk to each other, perform the following configuration:

```
vrf definition VRF3
rd 300:1
address-family ipv4
route-target export RT-EVPN-3 stitching
route-target import RT-EVPN-1 stitching
route-target import RT-EVPN-2 stitching
route-target export RT-3
route-target import RT-4
vrf definition VRF4
rd 400:1
address-family ipv4
route-target export RT-EVPN-4 stitching
route-target import RT-EVPN-1 stitching
route-target import RT-EVPN-2 stitching
route-target import RT-3
route-target export RT-4
```

Note For the above-mentioned use case, the CSR 1000v instance must configure EVPN on both VRF3 and VRF4.

Even IP BGP already imports all the routes from VRF3 and VRF4, BGP does not advertise the imported routes of the VRF to the EVPN peer.

You need to use the **Stitching** keyword in the configuration only when the sharing happens across the EVPN.

Configuring VRF Route Sharing

Procedure

Perform the following configuration to configure VRF Route Sharing in a hybrid cloud where VRF 1 and VRF 2 (On-Premise) can talk to to VRF 3 and VRF 4 (in the public cloud). In this sample solution, VRF3 and VRF4 cannot talk to each other.

Example:

```
vrf definition vrf3
rd 3:3
address-family ipv4
Route-target export 100:3
Route-target import 100:4
route-target export 3:3 stitching
route-target import 1:1 stitching
route-target import 2:2 stitching
exit-address-family
!
!
vrf definition vrf4
rd 4:4
address-family ipv4
Route-target import 100:3
Route-target export 100:4
route-target export 4:4 stitching
route-target import 1:1 stitching
route-target import 2:2 stitching
exit-address-family
!
!
interface BDI100
no shutdown
vrf forwarding vrf3
ip address 10.1.1.1 255.255.255.224
!
interface GigabitEthernet4.2
encapsulation dot1Q 2
vrf forwarding vrf3
ip address 4.4.4.1 255.255.255.224
bridge-domain 100
member vni 10100
!
interface nve1
source-interface loopback0
host-reachability protocol bgp
member vni 10100 vrf vrf3
!
router bgp 100
bgp router-id 11.11.11.11
no bgp default ipv4-unicast
neighbor 22.22.22.22 remote-as 200
neighbor 22.22.22.22 update-source loopback0
neighbor 22.22.22.22 ebgp-multihop 255
address-family ipv4 vrf vrf3
redistribute connected
neighbor 4.4.4.2 remote-as 300
neighbor 4.4.4.2 activate
neighbor 4.4.4.2 send-community both
advertise l2vpn evpn
exit-address-family
!
address-family l2vpn evpn
neighbor 22.22.22.22 activate
neighbor 22.22.22.22 send-community both
exit-address-family
end
```

Verifying VRF Route Sharing

Procedure

Step 1 **show ip bgp l2vpn evpn summary**.

Provides the BGP summary information for the VRF default address family (L2VPN EVPN).

Example:

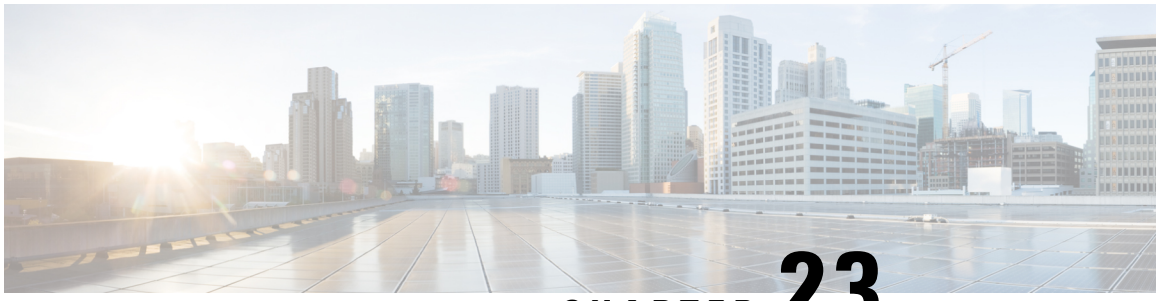
```
show ip bgp l2vpn evpn summary
BGP router identifier 11.11.11.11, local AS number 100
BGP table version is 8, main routing table version 8
7 network entries using 2408 bytes of memory
......
BGP activity 14/0 prefixes, 16/0 paths, scan interval 60 secs
7 networks peaked at 17:34:38 Aug 14 2019 CST (00:00:26.895 ago)
Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
22.22.22.22     4          200       6       5        4    0    0 00:01:23        4
Device#
```

Step 2 **show ip route vrf vrf3 bgp | in binding**.

Displays the IP routing table information associated with the VRF. When you see the output with the binding label, it indicates that the configuration is successful and BGP uses the binding label as the next hop.

Example:

```
+++ 17:35:05 Minuet(default) exec +++
show ip route vrf vrf3 bgp | in binding
B        10.2.1.0/24 [20/0] via binding label: 0x3000001, 00:00:26
B        10.2.2.0/24 [20/0] via binding label: 0x3000002, 00:00:26
B     192.168.1.0/24 [20/0] via binding label: 0x3000001, 00:00:26
B     192.168.2.0/24 [20/0] via binding label: 0x3000002, 00:00:26
Device#
```



CHAPTER 23

Troubleshooting Cisco CSR 1000v VM Issues

Verifying the Cisco CSR 1000v Hardware and VM Requirements

To help troubleshoot issues with the Cisco CSR 1000v, make sure that the router is installed on supported hardware and that the VM requirements are being met:

- Verify that the server hardware is supported by the hypervisor vendor.

  If using VMware, verify that the server is listed on the VMware Hardware Compatibility List. See the VMware documentation for more information.

- Verify that the I/O devices (for example, FC, iSCSI, SAS) being used are supported by the VM vendor.
- Verify that sufficient RAM is allocated on the server for the VMs and the hypervisor host.

  If using VMware, make sure the server has enough RAM to support both the VMs and VMware ESXi.

- Verify the hypervisor version is supported by the Cisco CSR 1000v.
- Verify that the correct VM settings for the amount of memory, number of CPUs, and disk size are configured.
- Verify that the vNICs are configured using a supported network driver. See Installation Overview.

Also see Cisco CSR 1000v Release Notes.

Troubleshooting Network Connectivity Issues

To troubleshoot network connectivity issues for the Cisco CSR 1000v, do the following:

- Verify that there is an active and unexpired license installed on the VM.

Enter the **show license** command. The License State should be shown as “Active, In Use”.

- Verify that the vNIC for the VMs are connected to the correct physical NIC, or to the proper vSwitch.
- If using virtual LANS (VLANs), make sure the vSwitch is configured with the correct VLAN.
- If using static MAC addresses, or VMs that are cloned, make sure there are no duplicate MAC addresses.

  Duplicate MAC addresses can cause the Cisco CSR 1000v feature license to become invalidated, which will disable the router interfaces.

Troubleshooting VM Performance Issues

The Cisco CSR 1000v operates within a set of supported VM parameters and settings to provide certain levels of performance that have been tested by Cisco. Use the vSphere Client to view data to troubleshoot VM performance. If you are using vCenter, you can view historical data. If you are not using vCenter, you can view live data from the host.

This is a list of troubleshooting tips for performance issues:

Troubleshooting—MTU

Verify that the router has the correct setting for maximum MTU.

By default, the maximum MTU on the router is 1500. To support jumbo frames, edit the default VMware vSwitch settings. For more information, see the VMware vSwitch documentation.

Note ESXi 5.0 supports a maximum MTU of 9000, even if jumbo frames are enabled on the router.

Troubleshooting—Memory

The Cisco CSR 1000v does not support memory sharing between VMs. On the ESXi host, check the memory counters to find out how much used memory and shared memory is on the VM. Verify that the balloon and swap used counters are zero.

If a given VM does not have enough memory to support the Cisco CSR 1000v, increase the size of the VM's memory. Insufficient memory on the VM or the host can cause the Cisco CSR 1000v console to hang and be non-responsive.

Note When troubleshooting performance issues, note that other VMs on the same host as the Cisco CSR 1000v can impact the performance of the Cisco CSR 1000v VM. Verify that other VMs on the host are not causing memory issues that are impacting the Cisco CSR 1000v VM.

Troubleshooting—Network Packets

Verify that no network packets are being dropped. On the ESXi host, check the network performance and view the counters to measure the number of receive packets and transmit packets dropped.

Troubleshooting—Throughput

Verify the current maximum throughput level with the **show platform hardware throughput level** command.

Troubleshooting—Instruction Extensions

Some x86 processors support instruction extensions for performing certain cryptographic transforms. Using these instructions is more efficient than not using them. The Cisco CSR 1000v/ISRv detects at runtime if the instruction extensions are available and will use them if they are available. To determine if the extensions are available, enter the **show platform software system all** command. (See the example below.)

If the output shows that "Crypto Supported" is "No", then the Cisco CSR 1000v/ISRv may not exhibit the expected throughput. This is an issue with either the underlying physical hardware or the hypervisor. Check to see if the underlying physical hardware is capable of exposing the extensions and also check to see if the hypervisor can expose the extensions.

If the output shows that "Crypto Supported" is "Yes", then the Cisco CSR 1000v/ISRv should provide the expected throughput, because the physical hardware and the hypervisor can expose the extensions.

In the following example, "Crypto Supported" is "Yes". Therefore the cryptographic transforms can use instruction extensions, and perform efficiently.

```
CSR1# show platform software system all
Processor Details
=================
Number of Processors : 4
Processor : 1 - 4
vendor_id : GenuineIntel
cpu MHz  : 3192.307
cache size : 20480 KB
Crypto Supported : Yes
```

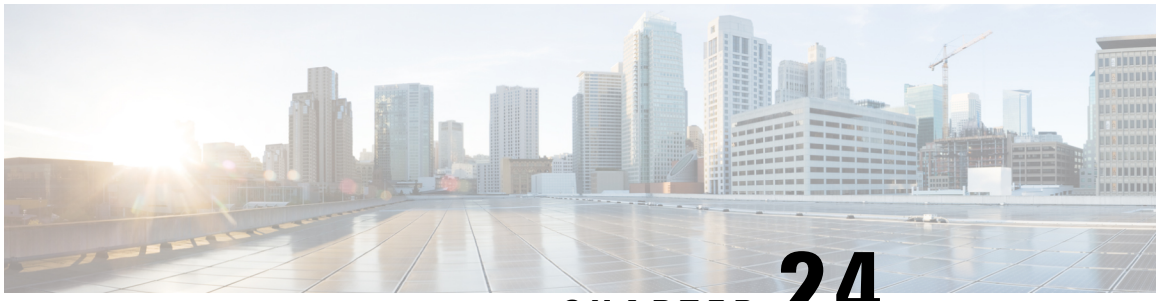
IP address Inconsistency Issues on the vSphere Web Client

As a user who's using CSR 1000v running on IOS XE 16.9.1 release or later, you might face inconsistencies in the IP addresses that is configured on the router and what is shown on the vSphere Web Client. At this moment there are no resolutions for this issue. See the following list to know why these inconsistencies might occur:

- ipv4 addresses for interfaces that are up or down are detected, while ipv6 addresses are only detected for interfaces that are up.
- After you perform an Interface Hot Delete, the vSphere Web Client continues to display the IP Address of the deleted interface.
- When you perform a reload on a CSR 1000v with addresses configured but not written to memory, the vSphere Web client continues to display the addresses even after the router comes up again. This occurs even though there are no addresses configured on the router. For example, configure Loopback, port-channel, port-group, and subinterfaces on a CSR 1000v router so that 63 addresses are displayed by the vSphere Web Client. Do not write the configuration to memory and reload the CSR 1000v. After the reload completes, all the 63 addresses are displayed on the Web Client. This occurs even though no addresses are configured on the CSR 1000v router. You can resolve this issue by configuring an address

on the CSR 1000v router. When you do so, the web client then removes the 63 address and just displays the newly configured address.

- When you configure multiple ipv6 addresses on an interface, only the last address that you configured is detected. If you unconfigure that address, none of the remaining configured ipv6 address on that interface are detected. This creates a state with multiple ipv6 addresses configured on an interface, but none displayed by the Web Client.
- When you delete interfaces, some of the addresses of the new interfaces are not displayed. This happens when the maximum number of IP Addresses are displayed and then you delete interfaces. For example, configure 32 Loopack interfaces with addresses and then delete each interface. Then, configure 32 GigabitEthernet subinterfaces with addresses. The addresses for the subinterfaces are not detected. This is because the router maintains entries for the deleted Loopback interfaces and is not able to add new interfaces.
- Addresses are detected for GigabitEthernet, Loopback, PortChannel, and VirtualPort-Group Interfaces as well as subinterfaces. However, Tunnel interface addresses are not detected.
- Secondary IP Addresses for IPv4 interfaces are not detected



CHAPTER 24

Rehosting the Cisco CSR 1000v License

Voluntarily Rehosting the License to a New VM

The process for rehosting a license on the Cisco CSR 1000v is different compared to other Cisco platforms. Because the license is not mapped to a Cisco hardware device, additional steps may be necessary for rehosting the license.

If you plan to voluntarily rehost the Cisco CSR 1000v to a new VM and the router is operating properly, you can use the self-service rehosting process on the Cisco Software Licensing Tool.

Note The self-service rehosting process is only available for permanent licenses on the Cisco CSR 1000v. If you have subscription term licenses installed, you must contact licensing@cisco.com for assistance.

SUMMARY STEPS

1. Access the Cisco Software Licensing portal
2. Click **Continue to Product License Registration**.
3. On the Product License Registration page, choose **Transfer > License for Transfer - Initiate**.
4. Specify the Source License.
5. Specify the Target and Options for the rehost license.
6. Review the license rehost information for accuracy. If the license information is valid, click **Submit**.

DETAILED STEPS

Procedure

Step 1 Access the Cisco Software Licensing portal

Step 2 Click **Continue to Product License Registration**.

Step 3 On the Product License Registration page, choose **Transfer > License for Transfer - Initiate**.

See the "License to Initiate Transfer" figure below.

Figure 18: License to Initiate-Transfer Screen

Step 4 Specify the Source License.

Select the license with the original node-locked UDI for your system. See the "Source Rehost License" figure below .

Note
If you changed the virtual UDI on the Cisco CSR 1000v using the **request license new-udi** command, the original node-locked UDI is invalidated on the router. Use the **show license udi history** command to obtain the node-locked UDI for your license that is stored in the Cisco Software Licensing Tool records. You can also verify the original node-locked UDI with the Cisco email confirmation you received when the license was purchased.

Click **Next**.

Figure 19: Source Rehost License

Step 5 Specify the Target and Options for the rehost license.

Click **Next**.

Note
When specifying the Target rehost license, use the new vUDI.

Step 6 Review the license rehost information for accuracy. If the license information is valid, click **Submit**.

See the "License Reshot Review" figure below.

Figure 20: License Rehost Review

The license portal processes the license request. You will receive an email confirming the new rehost licenses.

Obtaining a Rehost License if the System Fails

There may be cases when the Cisco CSR 1000v is not accessible due to a system failure and you need to rehost the existing licenses to a replacement device. Examples of a system failure may include:

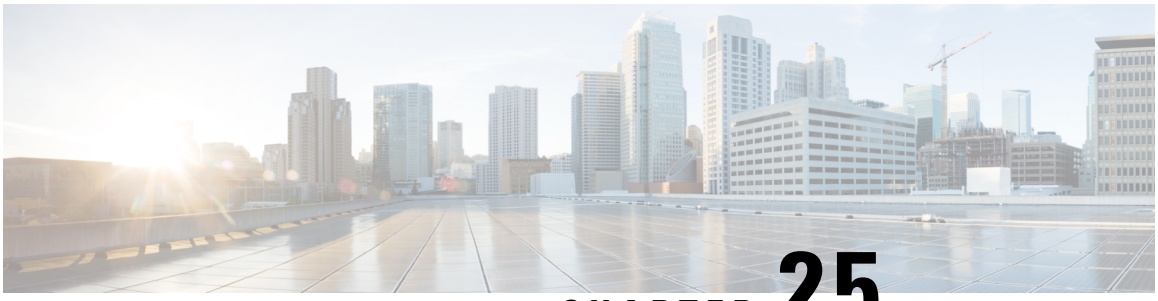
- The VM instance that the Cisco CSR 1000v was installed on was removed.
- The system server or host that the Cisco CSR 1000v VM instance was installed on experienced a hardware failure.

In this case, you need to obtain a rehost license and install it on a new VM. If you have a perpetual license, you can use the self-service rehosting process to obtain a rehost license.

Note The self-service rehosting process is only available for permanent licenses on the Cisco CSR 1000v. If you have subscription term licenses installed, you must contact licensing@cisco.com for assistance.

The following caveats apply if you are rehosting a perpetual license:

1. Do not select the **Transfer > License for RMA option. The RMA option does not support licenses for the Cisco CSR 1000v**. Use the **Transfer > License for Transfer - Initiate** option.
2. If you have the original Cisco license email confirmation with the original node-locked UDI, you can use the rehost option on the Cisco Software Licensing portal.
3. If you do not have the original Cisco license email confirmation with the original node-locked UDI, you must contact licensing@cisco.com for assistance. You will need to provide the PAK number from the original license purchase.
4. If you changed the virtual UDI on the Cisco CSR 1000v using the **request license new-udi** command and the VM is lost due to a system failure, the installed licenses will be destroyed. You must contact Cisco for assistance. You will need to the provide the PAK number from the original license purchase.



CHAPTER 25

Using the Web User Interface

Web User Interface Management

You can access your router using a web user interface, which allows you to monitor router performance using an easy-to-read graphical interface.

Note To manage and configure crypto map tunnels use CLI. The tunnels can also be configured with Virtual Tunnel Interface (VTI) and then the tunnels can be created with CLI and GUI.

You can basically configure a router by performing the steps in one of the following tasks:

Setting Up Factory Default Device Using WebUI

Quick Setup Wizard allows you to perform the basic router configuration. To configure the router:

Before you begin

- Before you access the WebUI, you need to have the basic configuration on the device.

Procedure

Step 1 Connect the RJ-45 end of a serial cable to the RJ-45 console port on the router.

Step 2 After the device initial configuration wizard appears, enter **No** to get into the device prompt when the following system message appears on the router.

Would you like to enter the initial configuration dialog? [yes/no]: no

Step 3 From the configuration mode, enter the following configuration parameters.

```
!
ip dhcp pool WEBUIPool
network 192.168.1.0 255.255.255.0
```

```
default-router 192.168.1.1
username webui privilege 15 password cisco
!
interface gig 0/0/1
ip address 192.168.1.1 255.255.255.0
!
```

Step 4 Connect your device to the router using an Ethernet cable to the gig 0/0/1 interface.

Step 5 Set up your system as a DHCP client to obtain the IP address of the router automatically.

Step 6 Launch the browser and enter the device IP address in your browser's address line. For a secure connection, type https://192.168.1.1/#/dayZeroRouting. For a less secure connection, enter http://192.168.1.1/#/dayZeroRouting.

Step 7 Enter the default username (webui) and default password (cisco).

Using Basic or Advanced Mode Setup Wizard

To configure the router using the basic or advanced mode setup:

Procedure

Step 1 Choose the **Basic Mode** or **Advanced Mode** and click **Go To Account Creation Page**.

Step 2 Enter the username and password. Reenter the password to confirm.

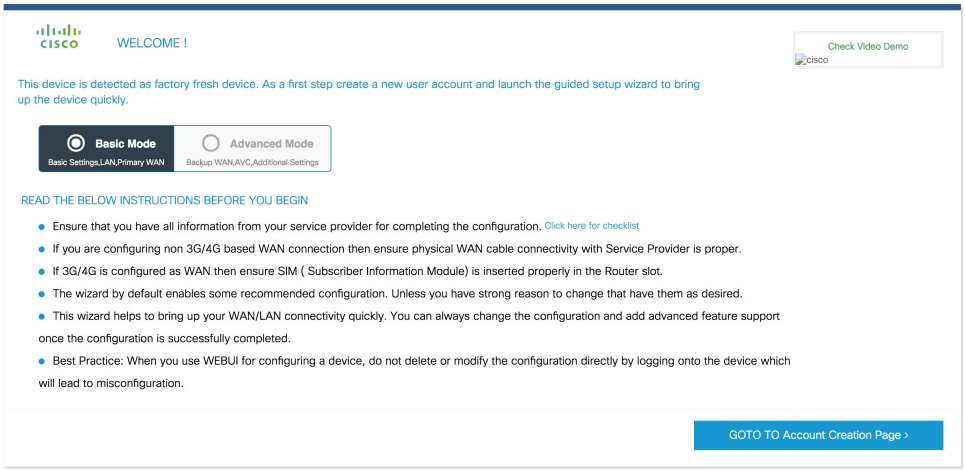
Step 3 Click **Create and Launch Wizard**.

Step 4 Enter the device name and domain name.

Step 5 Select the appropriate time zone from the **Time Zone** drop-down list.

Step 6 Select the appropriate date and time mode from the **Date and Time** drop-down list.

Step 7 Click **LAN Settings**.



Configure LAN Settings

Procedure

Step 1 Choose the **Web DHCP Pool/DHCP Pool** name or the **Create and Associate Access VLAN** option.

a) If you choose the Web DHCP Pool, specify the following:

Pool Name—Enter the DHCP Pool Name.

Network—Enter network address and the subnet mask.

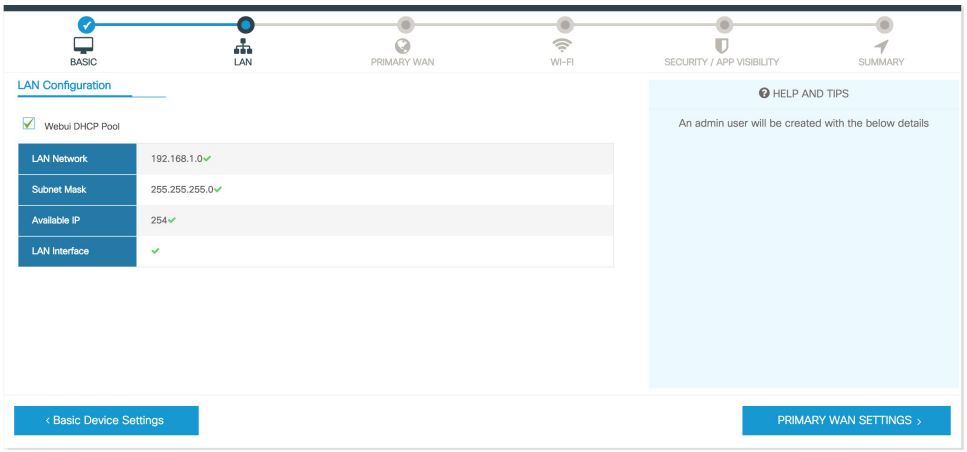
b) If you choose the Create and Associate Access VLAN option, specify the following:

Access VLAN—Enter the Access VLAN identification number. The range is from 1 to 4094.

Network—Enter the IP address of the VLAN.

Management Interfaces—Select the interface and move to the selected list box using the right and left arrows. You can also double click or drag and drop to move the interface to the selected list box.

Step 2 Click **Primary WAN Settings**.



Configure Primary WAN Settings

Procedure

Step 1 Select the primary WAN type. You can configure Serial, 3G/4G, Ethernet, or Broadband (xDSL) as primary WAN depending on the WAN types supported by the router.

Step 2 Select the interface from the drop-down list.

Step 3 Check the **Get DNS Server info directly from ISP** check box to get the DNS server information directly from the service provider. You can also manually enter the Primary DNS and Secondary DNS.

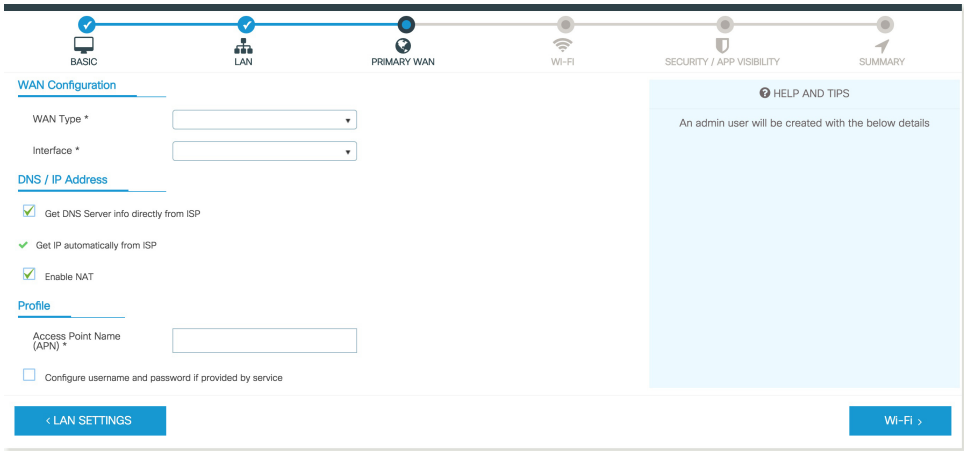
Step 4 Check the **Get IP automatically from ISP** check box to get the IP address information directly from the service provider. You can also manually enter the IP address and subnet mask.

Step 5 Check the **Enable NAT** check box to enable NAT. It is recommended to enable NAT.

Step 6 Check the **Enable PPPOE** check box to enable PPPoE. If you have enabled PPPoE, select the required authentication mode. The options are: **PAP** and **CHAP**.

Step 7 Enter the username and password provided by the service provider.

Step 8 Click **Security / APP Visibility WAN Settings**.



Configure Secondary WAN Settings

For advanced configuration, you should configure the secondary WAN connection.

Procedure

Step 1 Select the secondary WAN type. You can configure Serial, 3G/4G, Ethernet, or Broadband (xDSL) as a secondary WAN depending on the WAN types supported by the router.

Step 2 Select the interface from the drop-down list.

Step 3 Check the **Get DNS Server info directly from ISP** check box to get the DNS server information directly from the service provider. You can also manually enter the Primary DNS and Secondary DNS.

Step 4 Check the **Get IP automatically from ISP** check box to get the IP address information directly from the service provider. You can also manually enter the IP address and subnet mask.

Step 5 Check the **Enable NAT** check box to enable NAT. It is recommended to enable NAT.

Step 6 Check the **Enable PPPOE** check box to enable PPPoE. If you have enabled PPPoE, select the required authentication mode. The options are **PAP** and **CHAP**.

Step 7 Enter the username and password provided by the service provider.

Step 8 Click **Security / APP Visibility WAN Settings**.

Configure Security Settings

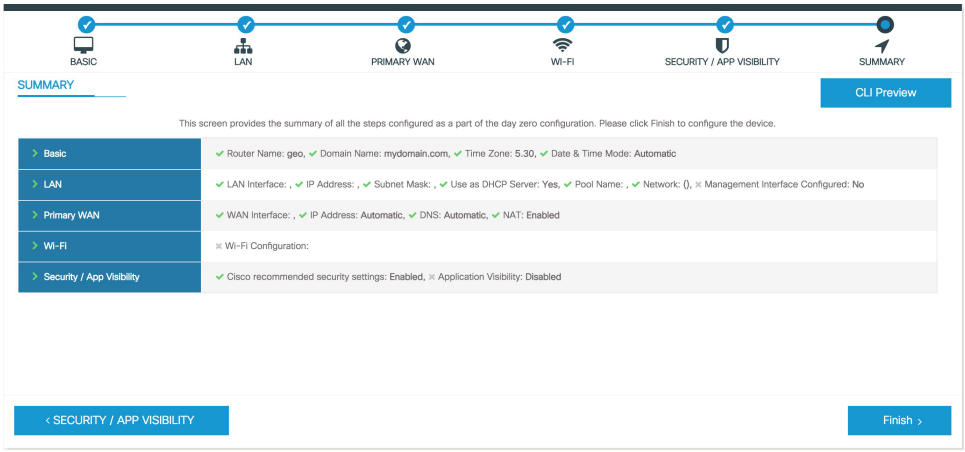
Procedure

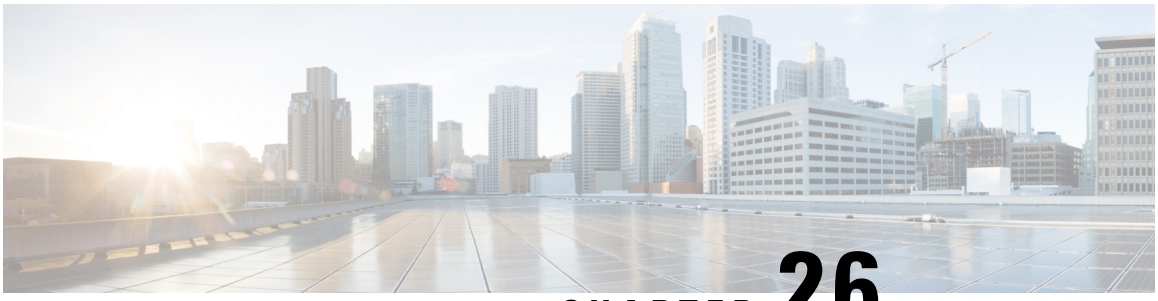
Step 1 Check the **Enable Cisco Recommended Security Settings** check box to ensure that all passwords are not shown in plain text. The passwords are encrypted.

Step 2 Click **Day 0 Config Summary**.

Step 3 To preview the configuration, click **CLI Preview** to preview the configuration.

Step 4 Click **Finish** to complete the Day Zero setup.





CHAPTER 26

Packet Trace

First Published: August 03, 2016

The Packet-Trace feature provides a detailed understanding of how data packets are processed by the Cisco IOS XE platform, and thus helps customers to diagnose issues and troubleshoot them more efficiently. This module provides information about how to use the Packet-Trace feature.

Information About Packet Trace

The Packet-Trace feature provides three levels of inspection for packets: accounting, summary, and path data. Each level provides a detailed view of packet processing at the cost of some packet processing capability. However, Packet Trace limits inspection to packets that match the debug platform condition statements, and is a viable option even under heavy-traffic situations in customer environments.

The following table explains the three levels of inspection provided by packet trace.

Table 54: Packet-Trace Level

Packet-Trace Level	Description
Accounting	Packet-Trace accounting provides a count of packets that enter and leave the network processor. Packet-Trace accounting is a lightweight performance activity, and runs continuously until it is disabled.
Summary	At the summary level of packet trace, data is collected for a finite number of packets. Packet-Trace summary tracks the input and output interfaces, the final packet state, and punt, drop, or inject packets, if any. Collecting summary data adds to additional performance compared to normal packet processing, and can help to isolate a troublesome interface.

Packet-Trace Level	Description
Path data	The packet-trace path data level provides the greatest level of detail in packet trace. Data is collected for a finite number of packets. Packet-Trace path data captures data, including a conditional debugging ID that is useful to correlate with feature debugs, a timestamp, and also feature-specific path-trace data. Path data also has two optional capabilities: packet copy and Feature Invocation Array (FIA) trace. The packet-copy option enables you to copy input and output packets at various layers of the packet (layer 2, layer 3 or layer 4). The FIA- trace option tracks every feature entry invoked during packet processing and helps you to know what is happening during packet processing. **Note** Collecting path data consumes more packet-processing resources, and the optional capabilities incrementally affect packet performance. Therefore, path-data level should be used in limited capacity or in situations where packet performance change is acceptable.

Usage Guidelines for Configuring Packet Trace

Consider the following best practices while configuring the Packet-Trace feature:

- Use of ingress conditions when using the Packet-Trace feature is recommended for a more comprehensive view of packets.
- Packet-trace configuration requires data-plane memory. On systems where data-plane memory is constrained, carefully consider how you will select the packet-trace values. A close approximation of the amount of memory consumed by packet trace is provided by the following equation:

memory required = (statistics overhead) + number of packets * (summary size + data size + packet copy size).

When the Packet-Trace feature is enabled, a small, fixed amount of memory is allocated for statistics. Similarly, when per-packet data is captured, a small, fixed amount of memory is required for each packet for summary data. However, as shown by the equation, you can significantly influence the amount of memory consumed by the number of packets you select to trace, and whether you collect path data and copies of packets.

Configuring Packet Trace

Perform the following steps to configure the Packet-Trace feature.

Note The amount of memory consumed by the Packet-Trace feature is affected by the packet-trace configuration. You should carefully select the size of per-packet path data and copy buffers and the number of packets to be traced in order to avoid interrupting normal services. You can check the current data-plane DRAM memory consumption by using the **show platform hardware qfp active infrastructure exmem statistics** command.

SUMMARY STEPS

1. **enable**
2. **debug platform packet-trace packet** *pkt-num* **[fia-trace | summary-only] [circular] [data-size** *data-size***]**
3. **debug platform packet-trace {punt |inject|copy|drop|packet|statistics}**
4. **debug platform condition [ipv4 | ipv6] [interface** *interface***][access-list** *access-list -name* **|** *ipv4-address* **/** *subnet-mask* **|** *ipv6-address* **/** *subnet-mask***] [ingress | egress |both]**
5. **debug platform condition start**
6. **debug platform condition stop**
7. **show platform packet-trace {configuration | statistics | summary | packet {all |** *pkt-num***}}**
8. **clear platform condition all**
9. **exit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables the privileged EXEC mode. Enter your password if prompted.
Step 2	**debug platform packet-trace packet** *pkt-num* **[fia-trace | summary-only] [circular] [data-size** *data-size***]** **Example:** `Router# debug platform packet-trace packets 2048 summary-only`	Collects summary data for a specified number of packets. Captures feature path data by default, and optionally performs FIA trace. *pkt-num*—Specifies the maximum number of packets maintained at a given time. **fia-trace**—Provides detailed level of data capture, including summary data, feature-specific data. Also displays each feature entry visited during packet processing. **summary-only**—Enables the capture of summary data with minimal details. **circular**—Saves the data of the most recently traced packets. *data-size*—Specifies the size of data buffers for storing feature and FIA trace data for each packet in bytes. When very heavy packet processing is performed on packets, users can increase the size of the data buffers if necessary. The default value is 2048.
Step 3	**debug platform packet-trace {punt |inject|copy|drop|packet|statistics}** **Example:** `Router# debug platform packet-trace punt`	Enables tracing of punted packets from data to control plane.

	Command or Action	Purpose
Step 4	**debug platform condition [ipv4 | ipv6] [interface** *interface*]**[access-list** *access-list -name | ipv4-address / subnet-mask | ipv6-address / subnet-mask*] **[ingress | egress |both]** **Example:** `Router# debug platform condition interface g0/0/0 ingress`	Specifies the matching criteria for tracing packets. Provides the ability to filter by protocol, IP address and subnet mask, access control list (ACL), interface, and direction.
Step 5	**debug platform condition start** **Example:** `Router# debug platform condition start`	Enables the specified matching criteria and starts packet tracing.
Step 6	**debug platform condition stop** **Example:** `Router# debug platform condition start`	Deactivates the condition and stops packet tracing.
Step 7	**show platform packet-trace {configuration | statistics | summary | packet {all |** *pkt-num*}} **Example:** `Router# show platform packet-trace 14`	Displays packet-trace data according to the specified option. See {start cross reference} Table 21-1 {end cross reference} for detailed information about the **show** command options.
Step 8	**clear platform condition all** **Example:** `Router(config)# clear platform condition all`	Removes the configurations provided by the **debug platform condition** and **debug platform packet-trace** commands.
Step 9	**exit** **Example:** `Router# exit`	Exits the privileged EXEC mode.

Configure Packet Tracer with UDF offset

Perform these steps to configure the Packet-Trace UDF with offset:

Procedure

Step 1 **enable**

Example:

```
Device> enable
```

Enables privileged EXEC mode.

- Enter your password if prompted.

Step 2 **configure terminal**

Example:

```
Device# configure terminal
```

Enters global configuration mode.

Step 3 **udf** *udf name* **header** {**inner | outer**} {**13|14**} **offset** *offset-in-bytes* **length** *length-in-bytes*

Example:

```
Router(config)# udf TEST_UDF_NAME_1 header  inner l3 64 1

Router(config)# udf TEST_UDF_NAME_2 header  inner l4 77 2

Router(config)# udf TEST_UDF_NAME_3 header outer l3 65 1
Router(config)# udf TEST_UDF_NAME_4 header outer l4 67 1
```

Configures individual UDF definitions. You can specify the name of the UDF, the networking header from which offset, and the length of data to be extracted.

The **inner** or **outer** keywords indicate the start of the offset from the unencapsulated Layer 3 or Layer 4 headers, or if there is an encapsulated packet, they indicate the start of offset from the inner L3/L4.

The **length** keyword specifies, in bytes, the length from the offset. The range is from 1 to 2.

.

Step 4 **udf** *udf name* {**header | packet-start**} *offset-base offset length*

Example:

```
Router(config)# udf TEST_UDF_NAME_5 packet-start 120 1
```

- header—Specifies the offset base configuration.
- packet-start—Specifies the offset base from packet-start. packet-start" can vary depending on if packet-trace is for an inbound packet or outbound packet. If the packet-trace is for an inbound packet then the packet-start will be layer2. For outbound, he packet-start will be layer3.
- offset—Specifies the number of bytes offset from the offset base. To match the first byte from the offset base (Layer 3/Layer 4 header), configure the offset as 0.
- length—Specifies the number of bytes from the offset. Only 1 or 2 bytes are supported. To match additional bytes, you must define multiple UDFs.

Step 5 **ip access-list extended** {*acl-name* |*acl-num*}

Example:

```
Router(config)# ip access-list extended acl2
```

Enables extended ACL configuration mode. The CLI enters the extended ACL configuration mode in which all subsequent commands apply to the current extended access list. Extended ACLs control traffic by the comparison of the source and destination addresses of the IP packets to the addresses configured in the ACL.

Step 6 **ip access-list extended { deny | permit } udf udf-name value mask**

Example:

```
Router(config-acl)# permit ip any any udf TEST_UDF_NAME_5 0xD3 0xFF
```

Configures the ACL to match on UDFs along with the current access control entries (ACEs) . The bytes defined in ACL is 0xD3. Masks are used with IP addresses in IP ACLs to specify what should be permitted and denied.

Step 7 **debug platform condition [ipv4 | ipv6] [interface** *interface***] [access-list** *access-list -name* **|** *ipv4-address* **/** *subnet-mask* **|** *ipv6-address* **/** *subnet-mask***] [ingress | egress |both]**

Example:

```
Router# debug platform condition interface gi0/0/0 ipv4 access-list acl2 both
```

Specifies the matching criteria for tracing packets. Provides the ability to filter by protocol, IP address and subnet mask, access control list (ACL), interface, and direction.

Step 8 **debug platform condition start**

Example:

```
Router# debug platform condition start
```

Enables the specified matching criteria and starts packet tracing.

Step 9 **debug platform packet-trace packet** *pkt-num* **[fia-trace | summary-only] [circular] [data-size** *data-size***]**

Example:

```
Router# debug platform packet-trace packet 1024 fia-trace data-size 2048
```

Collects summary data for a specified number of packets. Captures feature path data by default, and optionally performs FIA trace.

pkt-num—Specifies the maximum number of packets maintained at a given time.

fia-trace—Provides detailed level of data capture, including summary data, feature-specific data. Also displays each feature entry visited during packet processing.

summary-only—Enables the capture of summary data with minimal details.

circular—Saves the data of the most recently traced packets.

data-size—Specifies the size of data buffers for storing feature and FIA trace data for each packet in bytes. When very heavy packet processing is performed on packets, users can increase the size of the data buffers if necessary. The default value is 2048.

Step 10 **debug platform packet-trace {punt | inject|copy | drop |packet | statistics}**

Example:

```
Router# debug platform packet-trace punt
```

Enables tracing of punted packets from data to control plane.

Step 11 **debug platform condition stop**

Example:

```
Router# debug platform condition start
```

Deactivates the condition and stops packet tracing.

Step 12 **exit**

Example:

```
Router# exit
```

Exits the privileged EXEC mode.

Displaying Packet-Trace Information

Use these **show** commands to display packet-trace information.

Table 55: show Commands

Command	Description
show platform packet-trace configuration	Displays packet trace configuration, including any defaults.
show platform packet-trace statistics	Displays accounting data for all the traced packets.
show platform packet-trace summary	Displays summary data for the number of packets specified.
show platform packet-trace {all | *pkt-num***} [decode]**	Displays the path data for all the packets or the packet specified. The **decode** option attempts to decode the binary packet into a more human- readable form.

Removing Packet-Trace Data

Use these commands to clear packet-trace data.

Table 56: clear Commands

Command	Description
clear platform packet-trace statistics	Clears the collected packet-trace data and statistics.
clear platform packet-trace configuration	Clears the packet-trace configuration and the statistics.

Configuration Examples for Packet Trace

This section provides the following configuration examples:

Example: Configuring Packet Trace

This example describes how to configure packet trace and display the results. In this example, incoming packets to Gigabit Ethernet interface 0/0/1 are traced, and FIA-trace data is captured for the first 128 packets. Also, the input packets are copied. The **show platform packet-trace packet 0** command displays the summary data and each feature entry visited during packet processing for packet 0.

```
Router>
enable
Router# debug platform packet-trace packet 128 fia-trace
Router# debug platform packet-trace punt
Router# debug platform condition interface g0/0/1 ingress
Router# debug platform condition start
Router#! ping to UUT
Router# debug platform condition stop
Router# show platform packet-trace packet 0
Packet: 0           CBUG ID: 9
Summary
  Input     : GigabitEthernet0/0/1
  Output    : GigabitEthernet0/0/0
  State     : FWD
  Timestamp
    Start   : 1819281992118 ns (05/17/2014 06:42:01.207240 UTC)
    Stop    : 1819282095121 ns (05/17/2014 06:42:01.207343 UTC)
Path Trace
  Feature: IPV4
    Source      : 192.0.2.1
    Destination : 192.0.2.2
    Protocol    : 1 (ICMP)
  Feature: FIA_TRACE
    Entry     : 0x8059dbe8 - DEBUG_COND_INPUT_PKT
    Timestamp : 3685243309297
  Feature: FIA_TRACE
    Entry     : 0x82011a00 - IPV4_INPUT_DST_LOOKUP_CONSUME
    Timestamp : 3685243311450
  Feature: FIA_TRACE
    Entry     : 0x82000170 - IPV4_INPUT_FOR_US_MARTIAN
    Timestamp : 3685243312427
  Feature: FIA_TRACE
    Entry     : 0x82004b68 - IPV4_OUTPUT_LOOKUP_PROCESS
    Timestamp : 3685243313230
  Feature: FIA_TRACE
    Entry     : 0x8034f210 - IPV4_INPUT_IPOPTIONS_PROCESS
    Timestamp : 3685243315033
  Feature: FIA_TRACE
    Entry     : 0x82013200 - IPV4_OUTPUT_GOTO_OUTPUT_FEATURE
    Timestamp : 3685243315787
  Feature: FIA_TRACE
    Entry     : 0x80321450 - IPV4_VFR_REFRAG
    Timestamp : 3685243316980
  Feature: FIA_TRACE
    Entry     : 0x82014700 - IPV6_INPUT_L2_REWRITE
    Timestamp : 3685243317713
  Feature: FIA_TRACE
```

```
    Entry     : 0x82000080 - IPV4_OUTPUT_FRAG
    Timestamp : 3685243319223
  Feature: FIA_TRACE
    Entry     : 0x8200e500 - IPV4_OUTPUT_DROP_POLICY
    Timestamp : 3685243319950
  Feature: FIA_TRACE
    Entry     : 0x8059aff4 - PACTRAC_OUTPUT_STATS
    Timestamp : 3685243323603
  Feature: FIA_TRACE
    Entry     : 0x82016100 - MARMOT_SPA_D_TRANSMIT_PKT
    Timestamp : 3685243326183

Router# clear platform condition all
Router# exit
```

Linux Forwarding Transport Service (LFTS) is a transport mechanism to forward packets punted from the CPP into applications other than IOSd. This example displays the LFTS-based intercepted packet destined for binos application.

```
Router# show platform packet-trace packet 10
Packet: 10     CBUG ID: 52
Summary
  Input  : GigabitEthernet0/0/0
  Output : internal0/0/rp:1
  State  : PUNT 55 (For-us control)
  Timestamp
    Start : 597718358383 ns (06/06/2016 09:00:13.643341 UTC)
    Stop : 597718409650 ns (06/06/2016 09:00:13.643392 UTC)
Path Trace
  Feature: IPV4
    Input  : GigabitEthernet0/0/0
    Output : <unknown>
    Source : 10.64.68.2
    Destination : 10.0.0.102
    Protocol : 17 (UDP)
      SrcPort : 1985
      DstPort : 1985
  Feature: FIA_TRACE
    Input  : GigabitEthernet0/0/0
    Output : <unknown>
    Entry  : 0x8a0177bc - DEBUG_COND_INPUT_PKT
    Lapsed time : 426 ns
  Feature: FIA_TRACE
    Input  : GigabitEthernet0/0/0
    Output : <unknown>
    Entry  : 0x8a017788 - IPV4_INPUT_DST_LOOKUP_CONSUME
    Lapsed time : 386 ns
  Feature: FIA_TRACE
    Input  : GigabitEthernet0/0/0
    Output : <unknown>
    Entry  : 0x8a01778c - IPV4_INPUT_FOR_US_MARTIAN
    Lapsed time : 13653 ns
  Feature: FIA_TRACE
    Input  : GigabitEthernet0/0/0
    Output : internal0/0/rp:1
    Entry  : 0x8a017730 - IPV4_INPUT_LOOKUP_PROCESS_EXT
    Lapsed time : 2360 ns
  Feature: FIA_TRACE
    Input  : GigabitEthernet0/0/0
    Output : internal0/0/rp:1
    Entry  : 0x8a017be0 - IPV4_INPUT_IPOPTIONS_PROCESS_EXT
    Lapsed time : 66 ns
  Feature: FIA_TRACE
    Input  : GigabitEthernet0/0/0
```

```
  Output : internal0/0/rp:1
  Entry  : 0x8a017bfc - IPV4_INPUT_GOTO_OUTPUT_FEATURE_EXT
  Lapsed time : 680 ns
Feature: FIA_TRACE
  Input  : GigabitEthernet0/0/0
  Output : internal0/0/rp:1
  Entry  : 0x8a017d60 - IPV4_INTERNAL_ARL_SANITY_EXT
  Lapsed time : 320 ns
Feature: FIA_TRACE
  Input  : GigabitEthernet0/0/0
  Output : internal0/0/rp:1
  Entry  : 0x8a017a40 - IPV4_VFR_REFRAG_EXT
  Lapsed time : 106 ns
Feature: FIA_TRACE
  Input  : GigabitEthernet0/0/0
  Output : internal0/0/rp:1
  Entry  : 0x8a017d2c - IPV4_OUTPUT_DROP_POLICY_EXT
  Lapsed time : 1173 ns
Feature: FIA_TRACE
  Input  : GigabitEthernet0/0/0
  Output : internal0/0/rp:1
  Entry  : 0x8a017940 - INTERNAL_TRANSMIT_PKT_EXT
  Lapsed time : 20173 ns
LFTS Path Flow: Packet: 10    CBUG ID: 52
  Feature: LFTS
  Pkt Direction: IN
  Punt Cause  : 55
       subCause : 0
```

Example: Using Packet Trace

This example provides a scenario in which packet trace is used to troubleshoot packet drops for a NAT configuration on a Cisco device. This example shows how you can effectively utilize the level of detail provided by the Packet-Trace feature to gather information about an issue, isolate the issue, and then find a solution.

In this scenario, you can detect that there are issues, but are not sure where to start troubleshooting. You should, therefore, consider accessing the Packet-Trace summary for a number of incoming packets.

```
Router# debug platform condition ingress
Router# debug platform packet-trace packet 2048 summary-only
Router# debug platform condition start
Router# debug platform condition stop
Router# show platform packet-trace summary
Pkt   Input             Output            State  Reason
0     Gi0/0/0           Gi0/0/0           DROP   402 (NoStatsUpdate)
1     internal0/0/rp:0  internal0/0/rp:0  PUNT   21  (RP<->QFP keepalive)
2     internal0/0/recycle:0  Gi0/0/0      FWD
```

The output shows that packets are dropped due to NAT configuration on Gigabit Ethernet interface 0/0/0, which enables you to understand that an issue is occurring on a specific interface. Using this information, you can limit which packets to trace, reduce the number of packets for data capture, and increase the level of inspection.

```
Router# debug platform packet-trace packet 256
Router# debug platform packet-trace punt
Router# debug platform condition interface Gi0/0/0
Router# debug platform condition start
Router# debug platform condition stop
```

```
Router# show platform packet-trace summary
Router# show platform packet-trace 15
Packet: 15          CBUG ID: 238
Summary
  Input     : GigabitEthernet0/0/0
  Output    : internal0/0/rp:1
  State     : PUNT 55  (For-us control)
  Timestamp
    Start   : 1166288346725 ns (06/06/2016 09:09:42.202734 UTC)
    Stop    : 1166288383210 ns (06/06/2016 09:09:42.202770 UTC)
Path Trace
  Feature: IPV4
    Input       : GigabitEthernet0/0/0
    Output      : <unknown>
    Source      : 10.64.68.3
    Destination : 10.0.0.102
    Protocol    : 17 (UDP)
      SrcPort   : 1985
      DstPort   : 1985
IOSd Path Flow: Packet: 15    CBUG ID: 238
  Feature: INFRA
    Pkt Direction: IN
    Packet Rcvd From CPP
  Feature: IP
    Pkt Direction: IN
    Source      : 10.64.68.122
    Destination : 10.64.68.255
  Feature: IP
    Pkt Direction: IN
    Packet Enqueued in IP layer
    Source      : 10.64.68.122
    Destination : 10.64.68.255
    Interface   : GigabitEthernet0/0/0
  Feature: UDP
    Pkt Direction: IN
    src         : 10.64.68.122(1053)
    dst         : 10.64.68.255(1947)
    length      : 48

Router#show platform packet-trace packet 10
Packet: 10          CBUG ID: 10
Summary
  Input     : GigabitEthernet0/0/0
  Output    : internal0/0/rp:0
  State     : PUNT 55  (For-us control)
  Timestamp
    Start   : 274777907351 ns (01/10/2020 10:56:47.918494 UTC)
    Stop    : 274777922664 ns (01/10/2020 10:56:47.918509 UTC)
Path Trace
  Feature: IPV4(Input)
    Input       : GigabitEthernet0/0/0
    Output      : <unknown>
    Source      : 10.78.106.2
    Destination : 10.0.0.102
    Protocol    : 17 (UDP)
      SrcPort   : 1985
      DstPort   : 1985

IOSd Path Flow: Packet: 10    CBUG ID: 10
  Feature: INFRA
    Pkt Direction: IN
Packet Rcvd From DATAPLANE
 Feature: IP
    Pkt Direction: IN
```

```
    Packet Enqueued in IP layer
    Source      : 10.78.106.2
    Destination : 10.0.0.102
    Interface   : GigabitEthernet0/0/0

  Feature: UDP
    Pkt Direction: IN DROP
    Pkt : DROPPED
    UDP: Discarding silently
    src        : 881 10.78.106.2(1985)
    dst        : 10.0.0.102(1985)
    length     : 60

Router#show platform packet-trace packet  12
Packet: 12          CBUG ID: 767
Summary
  Input     : GigabitEthernet3
  Output    : internal0/0/rp:0
  State     : PUNT 11  (For-us data)
  Timestamp
    Start   : 16120990774814 ns (01/20/2020 12:38:02.816435 UTC)
    Stop    : 16120990801840 ns (01/20/2020 12:38:02.816462 UTC)
Path Trace
  Feature: IPV4(Input)
    Input       : GigabitEthernet3
    Output      : <unknown>
    Source      : 10.1.1.1
    Destination : 10.1.1.2
    Protocol    : 6 (TCP)
      SrcPort   : 46593
      DstPort   : 23
IOSd Path Flow: Packet: 12    CBUG ID: 767
  Feature: INFRA
    Pkt Direction: IN
    Packet Rcvd From DATAPLANE

  Feature: IP
    Pkt Direction: IN
    Packet Enqueued in IP layer
    Source      : 10.1.1.1
    Destination : 10.1.1.2
    Interface   : GigabitEthernet3

  Feature: IP
    Pkt Direction: IN
    FORWARDEDTo transport layer
    Source        : 10.1.1.1
    Destination   : 10.1.1.2
    Interface     : GigabitEthernet3

  Feature: TCP
    Pkt Direction: IN
    tcp0: I NoTCB 10.1.1.1:46593 10.1.1.2:23 seq 1925377975 OPTS 4 SYN  WIN 4128

Router# show platform packet-trace summary
Pkt   Input                     Output                    State  Reason
0     INJ.2                     Gi1                       FWD
1     Gi1                       internal0/0/rp:0          PUNT   11  (For-us data)
2     INJ.2                     Gi1                       FWD
3     Gi1                       internal0/0/rp:0          PUNT   11  (For-us data)
4     INJ.2                     Gi1                       FWD
5     INJ.2                     Gi1                       FWD
6     Gi1                       internal0/0/rp:0          PUNT   11  (For-us data)
7     Gi1                       internal0/0/rp:0          PUNT   11  (For-us data)
8     Gi1                       internal0/0/rp:0          PUNT   11  (For-us data)
```

```
9     Gi1                       internal0/0/rp:0          PUNT   11  (For-us data)
10    INJ.2                     Gi1                       FWD
11    INJ.2                     Gi1                       FWD
12    INJ.2                     Gi1                       FWD
13    Gi1                       internal0/0/rp:0          PUNT   11  (For-us data)
14    Gi1                       internal0/0/rp:0          PUNT   11  (For-us data)
15    Gi1                       internal0/0/rp:0          PUNT   11  (For-us data)
16    INJ.2                     Gi1                       FWD
```

The following example displays the packet trace data statistics.

```
Router#show platform packet-trace statistics
Packets Summary
  Matched  3
  Traced   3
Packets Received
  Ingress  0
  Inject   0
Packets Processed
  Forward  0
  Punt     3
    Count       Code  Cause
    3           56    RP injected for-us control
  Drop     0
  Consume  0

          PKT_DIR_IN
             Dropped       Consumed       Forwarded
INFRA            0             0             0
TCP              0             0             0
UDP              0             0             0
IP               0             0             0
IPV6             0             0             0
ARP              0             0             0

         PKT_DIR_OUT
             Dropped       Consumed       Forwarded
INFRA            0             0             0
TCP              0             0             0
UDP              0             0             0
IP               0             0             0
IPV6             0             0             0
ARP              0             0             0
```

The following example displays packets that are injected and punted to the forwarding processor from the control plane.

```
Router#debug platform condition ipv4 10.118.74.53/32 both
 Router#Router#debug platform condition start
Router#debug platform packet-trace packet 200
Packet count rounded up from 200 to 256

Router#show platform packet-tracer packet 0
show plat pack pa 0
Packet: 0           CBUG ID: 674
Summary
  Input     : GigabitEthernet1
  Output    : internal0/0/rp:0
  State     : PUNT 11  (For-us data)
  Timestamp
    Start   : 17756544435656 ns (06/29/2020 18:19:17.326313 UTC)
    Stop    : 17756544469451 ns (06/29/2020 18:19:17.326346 UTC)
Path Trace
  Feature: IPV4(Input)
    Input       : GigabitEthernet1
```

```
    Output      : <unknown>
    Source      : 10.118.74.53
    Destination : 172.18.124.38
    Protocol    : 17 (UDP)
      SrcPort   : 2640
      DstPort   : 500

IOSd Path Flow: Packet: 0    CBUG ID: 674
  Feature: INFRA
  Pkt Direction: IN
    Packet Rcvd From DATAPLANE

  Feature: IP
  Pkt Direction: IN
    Packet Enqueued in IP layer
    Source      : 10.118.74.53
    Destination : 172.18.124.38
    Interface   : GigabitEthernet1

  Feature: IP
  Pkt Direction: IN
  FORWARDED To transport layer
    Source        : 10.118.74.53
    Destination   : 172.18.124.38
    Interface     : GigabitEthernet1

  Feature: UDP
  Pkt Direction: IN
  DROPPED
 UDP: Checksum error: dropping
 Source      : 10.118.74.53(2640)
 Destination : 172.18.124.38(500)

Router#show platform packet-tracer packet 2
Packet: 2           CBUG ID: 2

IOSd Path Flow:
  Feature: TCP
  Pkt Direction: OUTtcp0: O SYNRCVD 172.18.124.38:22 172.18.124.55:52774 seq 3052140910
OPTS 4 ACK 2346709419 SYN  WIN 4128

  Feature: TCP
  Pkt Direction: OUT
  FORWARDED
 TCP: Connection is in SYNRCVD state
 ACK         : 2346709419
 SEQ         : 3052140910
 Source      : 172.18.124.38(22)
 Destination : 172.18.124.55(52774)


  Feature: IP
  Pkt Direction: OUTRoute out the generated packet.srcaddr: 172.18.124.38, dstaddr:
172.18.124.55

  Feature: IP
  Pkt Direction: OUTInject and forward successful srcaddr: 172.18.124.38, dstaddr:
172.18.124.55

  Feature: TCP
  Pkt Direction: OUTtcp0: O SYNRCVD 172.18.124.38:22 172.18.124.55:52774 seq 3052140910
OPTS 4 ACK 2346709419 SYN  WIN 4128
Summary
  Input     : INJ.2
```

```
  Output    : GigabitEthernet1
  State     : FWD
  Timestamp
    Start   : 490928006866 ns (06/29/2020 13:31:30.807879 UTC)
    Stop    : 490928038567 ns (06/29/2020 13:31:30.807911 UTC)
Path Trace
  Feature: IPV4(Input)
    Input       : internal0/0/rp:0
    Output      : <unknown>
    Source      : 172.18.124.38
    Destination : 172.18.124.55
    Protocol    : 6 (TCP)
      SrcPort   : 22
      DstPort   : 52774
  Feature: IPSec
    Result    : IPSEC_RESULT_DENY
    Action    : SEND_CLEAR
    SA Handle : 0
    Peer Addr : 10.124.18.172
    Local Addr: 10.124.18.172


Router#
```

Example: Use packet trace

This example provides a scenario in which packet trace is used to troubleshoot packet drops for a NAT configuration on a Cisco ASR 1006 Router. This example shows how you can effectively utilize the level of detail provided by the Packet-Trace feature to gather information about an issue, isolate the issue, and then find a solution.

In this scenario, you can detect that there are issues, but are not sure where to start troubleshooting. You should, therefore, consider accessing the Packet-Trace summary for a number of incoming packets.

```
Router# debug platform condition ingress
Router# debug platform packet-trace packet 2048 summary-only
Router# debug platform condition start
Router# debug platform condition stop
Router# show platform packet-trace summary
Pkt   Input             Output            State  Reason
0     Gi0/0/0           Gi0/0/0           DROP   402 (NoStatsUpdate)
1     internal0/0/rp:0  internal0/0/rp:0  PUNT   21  (RP<->QFP keepalive)
2     internal0/0/recycle:0  Gi0/0/0      FWD
```

The output shows that packets are dropped due to NAT configuration on Gigabit Ethernet interface 0/0/0, which enables you to understand that an issue is occurring on a specific interface. Using this information, you can limit which packets to trace, reduce the number of packets for data capture, and increase the level of inspection.

```
Router# debug platform packet-trace packet 256
Router# debug platform packet-trace punt
Router# debug platform condition interface Gi0/0/0
Router# debug platform condition start
Router# debug platform condition stop
Router# show platform packet-trace summary
Router# show platform packet-trace 15
Packet: 15          CBUG ID: 238
Summary
  Input     : GigabitEthernet0/0/0
```

```
  Output    : internal0/0/rp:1
  State     : PUNT 55  (For-us control)
  Timestamp
    Start   : 1166288346725 ns (06/06/2016 09:09:42.202734 UTC)
    Stop    : 1166288383210 ns (06/06/2016 09:09:42.202770 UTC)
Path Trace
  Feature: IPV4
    Input       : GigabitEthernet0/0/0
    Output      : <unknown>
    Source      : 10.64.68.3
    Destination : 224.0.0.102
    Protocol    : 17 (UDP)
      SrcPort   : 1985
      DstPort   : 1985
IOSd Path Flow: Packet: 15    CBUG ID: 238
  Feature: INFRA
    Pkt Direction: IN
    Packet Rcvd From CPP
  Feature: IP
    Pkt Direction: IN
    Source      : 10.64.68.122
    Destination : 10.64.68.255
  Feature: IP
    Pkt Direction: IN
    Packet Enqueued in IP layer
    Source      : 10.64.68.122
    Destination : 10.64.68.255
    Interface   : GigabitEthernet0/0/0
  Feature: UDP
    Pkt Direction: IN
    src         : 10.64.68.122(1053)
    dst         : 10.64.68.255(1947)
    length      : 48

Router#show platform packet-trace packet 10
Packet: 10          CBUG ID: 10
Summary
  Input     : GigabitEthernet0/0/0
  Output    : internal0/0/rp:0
  State     : PUNT 55  (For-us control)
  Timestamp
    Start   : 274777907351 ns (01/10/2020 10:56:47.918494 UTC)
    Stop    : 274777922664 ns (01/10/2020 10:56:47.918509 UTC)
Path Trace
  Feature: IPV4(Input)
    Input       : GigabitEthernet0/0/0
    Output      : <unknown>
    Source      : 10.78.106.2
    Destination : 224.0.0.102
    Protocol    : 17 (UDP)
      SrcPort   : 1985
      DstPort   : 1985

IOSd Path Flow: Packet: 10    CBUG ID: 10
  Feature: INFRA
    Pkt Direction: IN
Packet Rcvd From DATAPLANE
 Feature: IP
    Pkt Direction: IN
    Packet Enqueued in IP layer
    Source      : 10.78.106.2
    Destination : 224.0.0.102
    Interface   : GigabitEthernet0/0/0
```

```
  Feature: UDP
    Pkt Direction: IN DROP
    Pkt : DROPPED
    UDP: Discarding silently
    src        : 881 10.78.106.2(1985)
    dst        : 224.0.0.102(1985)
    length     : 60

Router#show platform packet-trace packet  12
Packet: 12          CBUG ID: 767
Summary
  Input     : GigabitEthernet3
  Output    : internal0/0/rp:0
  State     : PUNT 11  (For-us data)
  Timestamp
    Start   : 16120990774814 ns (01/20/2020 12:38:02.816435 UTC)
    Stop    : 16120990801840 ns (01/20/2020 12:38:02.816462 UTC)
Path Trace
  Feature: IPV4(Input)
    Input       : GigabitEthernet3
    Output      : <unknown>
    Source      : 12.1.1.1
    Destination : 12.1.1.2
    Protocol    : 6 (TCP)
      SrcPort   : 46593
      DstPort   : 23
IOSd Path Flow: Packet: 12    CBUG ID: 767
  Feature: INFRA
    Pkt Direction: IN
    Packet Rcvd From DATAPLANE

  Feature: IP
    Pkt Direction: IN
    Packet Enqueued in IP layer
    Source      : 12.1.1.1
    Destination : 12.1.1.2
    Interface   : GigabitEthernet3

  Feature: IP
    Pkt Direction: IN
    FORWARDEDTo transport layer
    Source        : 12.1.1.1
    Destination   : 12.1.1.2
    Interface     : GigabitEthernet3

  Feature: TCP
    Pkt Direction: IN
    tcp0: I NoTCB 12.1.1.1:46593 12.1.1.2:23 seq 1925377975 OPTS 4 SYN  WIN 4128

Router# show platform packet-trace summary
Pkt   Input                     Output                    State  Reason
0     INJ.2                     Gi1                       FWD
1     Gi1                       internal0/0/rp:0          PUNT   11  (For-us data)
2     INJ.2                     Gi1                       FWD
3     Gi1                       internal0/0/rp:0          PUNT   11  (For-us data)
4     INJ.2                     Gi1                       FWD
5     INJ.2                     Gi1                       FWD
6     Gi1                       internal0/0/rp:0          PUNT   11  (For-us data)
7     Gi1                       internal0/0/rp:0          PUNT   11  (For-us data)
8     Gi1                       internal0/0/rp:0          PUNT   11  (For-us data)
9     Gi1                       internal0/0/rp:0          PUNT   11  (For-us data)
10    INJ.2                     Gi1                       FWD
11    INJ.2                     Gi1                       FWD
12    INJ.2                     Gi1                       FWD
13    Gi1                       internal0/0/rp:0          PUNT   11  (For-us data)
```

```
14    Gi1                       internal0/0/rp:0          PUNT   11  (For-us data)
15    Gi1                       internal0/0/rp:0          PUNT   11  (For-us data)
16    INJ.2                     Gi1                       FWD
```

The example displays the packet trace data statistics.

```
Router#show platform packet-trace statistics
Packets Summary
  Matched  3
  Traced   3
Packets Received
  Ingress  0
  Inject   0
Packets Processed
  Forward  0
  Punt     3
    Count       Code  Cause
    3           56    RP injected for-us control
  Drop     0
  Consume  0

          PKT_DIR_IN
             Dropped       Consumed       Forwarded
INFRA            0             0             0
TCP              0             0             0
UDP              0             0             0
IP               0             0             0
IPV6             0             0             0
ARP              0             0             0

         PKT_DIR_OUT
             Dropped       Consumed       Forwarded
INFRA            0             0             0
TCP              0             0             0
UDP              0             0             0
IP               0             0             0
IPV6             0             0             0
ARP              0             0             0
```

The example displays packets that are injected and punted to the forwarding processor from the control plane.

```
Router#debug platform condition ipv4 10.118.74.53/32 both
 Router#Router#debug platform condition start
Router#debug platform packet-trace packet 200
Packet count rounded up from 200 to 256

Router#show platform packet-tracer packet 0
show plat pack pa 0
Packet: 0           CBUG ID: 674
Summary
  Input     : GigabitEthernet1
  Output    : internal0/0/rp:0
  State     : PUNT 11  (For-us data)
  Timestamp
    Start   : 17756544435656 ns (06/29/2020 18:19:17.326313 UTC)
    Stop    : 17756544469451 ns (06/29/2020 18:19:17.326346 UTC)
Path Trace
  Feature: IPV4(Input)
    Input       : GigabitEthernet1
    Output      : <unknown>
    Source      : 10.118.74.53
    Destination : 198.51.100.38
    Protocol    : 17 (UDP)
      SrcPort   : 2640
      DstPort   : 500
```

```
IOSd Path Flow: Packet: 0    CBUG ID: 674
  Feature: INFRA
  Pkt Direction: IN
    Packet Rcvd From DATAPLANE

  Feature: IP
  Pkt Direction: IN
    Packet Enqueued in IP layer
    Source      : 10.118.74.53
    Destination : 198.51.100.38
    Interface   : GigabitEthernet1

  Feature: IP
  Pkt Direction: IN
  FORWARDED To transport layer
    Source        : 10.118.74.53
    Destination   : 198.51.100.38
    Interface     : GigabitEthernet1

  Feature: UDP
  Pkt Direction: IN
  DROPPED
 UDP: Checksum error: dropping
 Source      : 10.118.74.53(2640)
 Destination : 198.51.100.38(500)

Router#show platform packet-tracer packet 2
Packet: 2           CBUG ID: 2

IOSd Path Flow:
  Feature: TCP
  Pkt Direction: OUTtcp0: O SYNRCVD 198.51.100.38:22 198.51.100.55:52774 seq 3052140910
OPTS 4 ACK 2346709419 SYN  WIN 4128

  Feature: TCP
  Pkt Direction: OUT
  FORWARDED
 TCP: Connection is in SYNRCVD state
 ACK         : 2346709419
 SEQ         : 3052140910
 Source      : 198.51.100.38(22)
 Destination : 198.51.100.55(52774)


  Feature: IP
  Pkt Direction: OUTRoute out the generated packet.srcaddr: 198.51.100.38, dstaddr:
198.51.100.55

  Feature: IP
  Pkt Direction: OUTInject and forward successful srcaddr: 198.51.100.38, dstaddr:
198.51.100.55

  Feature: TCP
  Pkt Direction: OUTtcp0: O SYNRCVD 198.51.100.38:22 198.51.100.55:52774 seq 3052140910
OPTS 4 ACK 2346709419 SYN  WIN 4128
Summary
  Input     : INJ.2
  Output    : GigabitEthernet1
  State     : FWD
  Timestamp
    Start   : 490928006866 ns (06/29/2020 13:31:30.807879 UTC)
    Stop    : 490928038567 ns (06/29/2020 13:31:30.807911 UTC)
Path Trace
```

```
  Feature: IPV4(Input)
    Input       : internal0/0/rp:0
    Output      : <unknown>
    Source      : 172.18.124.38
    Destination : 172.18.124.55
    Protocol    : 6 (TCP)
      SrcPort   : 22
      DstPort   : 52774
  Feature: IPSec
    Result    : IPSEC_RESULT_DENY
    Action    : SEND_CLEAR
    SA Handle : 0
    Peer Addr : 55.124.18.172
    Local Addr: 38.124.18.172


Router#
```

Additional References

Standards

Standard	Title
None	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at this URL: {start hypertext}http://www.cisco.com/go/mibs{end hypertext}

RFCs

RFC	Title
None	—

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	{start hypertext}http://www.cisco.com/cisco/web/support/index.html{end hypertext}

Feature Information for Packet Trace

{start cross reference}Table 21-4{end cross reference} lists the features in this module and provides links to specific configuration information.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to{start hypertext} http://www.cisco.com/go/cfn{end hypertext}. An account on Cisco.com is not required.

Note {start cross reference}Table 21-4{end cross reference} lists only the software releases that support a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 57: Feature Information for Packet Trace

Feature Name	Releases	Feature Information
Packet Trace	Cisco IOS XE 3.10S	The Packet Trace feature provides information about how data packets are processed by the Cisco IOS XE software. In Cisco IOS XE Release 3.10S, this feature was introduced. The following commands were introduced or modified: • **debug platform packet-trace packet** *pkt-num* [**fia-trace** | **summary-only**] [**data-size** *data-size*] [**circular**] • **debug platform packet-trace copy packet** {**input** | **output** | **both**} [**size** *num-bytes*] [**L2** | **L3** | **L4**] • **show platform packet-trace** {**configuration** | **statistics** | **summary** | **packet** {**all** | *pkt-num*}}
	Cisco IOS XE 3.11S	In Cisco IOS XE Release 3.11S, this feature was enhanced to include the following features: • Matched versus traced statistics. • Trace stop timestamp in addition to trace start timestamp. The following commands were introduced or modified: • **debug platform packet-trace drop** [**code** *drop-num*] • **show platform packet-trace packet** {**all** | *pkt-num*} [*decode*]
	Cisco IOS XE Denali 16.3.1	In Cisco IOS XE Denali 16.3.1, this feature was enhanced to include Layer3 packet tracing along with IOSd. The following commands were introduced or modified: **debug platform packet-trace punt**.
	Cisco IOS XE Amsterdam 17.3.1	The output of the **show platform packet-trace** command now includes additional trace information for packets either originated from IOSd or destined to IOSd or other BinOS processes.