



Cisco UCS C-Series IMC Emulator Quick Start Guide

[Cisco IMC Emulator](#) **2**

[Overview](#) **2**

[Setting up Cisco IMC Emulator](#) **3**

[Using Cisco IMC Emulator](#) **8**

Revised: May 26, 2017,

Cisco IMC Emulator

Overview

About Cisco IMC Emulator

The Cisco IMC Emulator application emulates the Cisco Integrated Management Controller (Cisco IMC) on select UCS C-series servers. The emulator allows users to familiarize themselves with Cisco's management software and hardware offerings, without the need for a physical server.

The Cisco IMC Emulator consists of a platform-specific Cisco IMC image and a machine emulator for BMC controller, DRAM, flash, and network. The emulator can be run on a Windows or a 64-bit Linux computer.

Use the Cisco IMC Emulator to

- View the available Cisco IMC features before upgrading to a new release.
- Set up your software and test the communication with Cisco IMC without having to wait for the real hardware.

Supported Servers

Cisco IMC Emulator is available for the following servers:

- UCS C460 M4
- UCS C220 M4
- UCS C240 M4



Note Users cannot run all the server-specific emulators simultaneously on a Linux host. Only one server-specific emulator can be run on the Linux host.

Supported FRUs

The following FRUs are supported in the emulator:

- Front Panel FRU
- Motherboard FRU
- Power Supply 1 FRU
- Power Supply 2 FRU
- VIC FRU
- MLOM FRU

- TPM FRU

Supported Inventories

The following inventories are currently supported in the emulator:

- CPU
- Memory
- Power Supply
- PCI Adapters
- VIC Adapters
- Storage Adapters

Limitations

Cisco IMC Emulator has the following limitations:

- Supports only a dedicated network mode.
- Does not support the host features, as it does not emulate the host machine. For example, KVM or virtual media cannot be accessed through the emulator.
- Does not support configuration for storage or flex flash cards.
- Does not support firmware updates.

Setting up Cisco IMC Emulator

Prerequisites

Supported Operating Systems

Cisco IMC Emulator runs on the following operating systems:

- Red Hat Enterprise Linux 7.0
- CentOS 6.5

Prerequisites for a Linux Computer

- You require a 64-bit Linux host to run the emulator.
- You must have root privileges to start the emulator.
- Create a TAP or a bridge network on the Linux host to communicate with the emulator.

The host machine requires the **tunctl** and **brctl** commands to create the tap network and the bridge network, respectively.

- Make sure Python 2.6 or later has been installed on your computer. (Python is required to start the IMC Emulator Configuration UI.)
- Disable the firewall on the Linux host to access the emulator externally.
- Enable hardware virtualization in the BIOS setup of the Linux host.
See [Enabling Hardware Virtualization in BIOS, on page 4](#).

Prerequisites for a Windows Computer

- Install Oracle VirtualBox on the Windows computer.
- Install the 64-bit Linux operating system in the VirtualBox software.
- Create a TAP or a bridge network mode in VirtualBox.
Enable bridge network mode in Virtual Box settings to access the emulator externally. The bridge mode works only with a wired network (LAN cable) in VirtualBox.
- Make sure Python 2.6 or later has been installed on your computer. (Python is required to start the IMC Emulator Configuration UI.)
- Disable the firewall on the Linux host to access the emulator externally.
- Enable hardware virtualization in the BIOS setup of the Windows computer.
See [Enabling Hardware Virtualization in BIOS, on page 4](#).

Enabling Hardware Virtualization in BIOS

While most PCs support hardware virtualization, this feature might not have been enabled by default. The procedure to enable virtualization might vary depending on your computer's motherboard, processor type, chipset and OEM. For the correct information on configuring your system, refer to the documentation for your computer.

- Restart the computer and open the **BIOS** menu.
To open the **BIOS** menu, press **F4** or **F8** or any other appropriate key depending on your computer.
- Open the **Processor** submenu.
- Enable **Virtualization**. The setting might be called **Intel VT-x** or **AMD-V**.
- Enable **Intel VT-d** or **AMD IOMMU**, if the options are available.
- Save the setting and exit the **BIOS** menu.

Extracting the Release Package

The release package consists of the following files:

- Server-specific Cisco IMC Image
Cisco IMC Emulator supports the C220 M4, C240 M4, and C460 M4 servers.
- Launch.sh—script file to start the emulator configuration Web UI
Run this script with root privileges.

- network.sh—script file to create a TAP or a bridge network using the CLI
- UCSC-CXXX-M4-Config.ini—sensor configuration file
- Binary files

The binary files such as FRU, SMBIOS, and DIMM SPD are extracted from the physical server.

To extract the **cimc_emulator-x.x.x.x.zip** release package, run the following command:

```
$unzip cimc_emulator-x.x.x.x.zip
```

Guidelines For Creating a Bridge or TAP Network on the Host

To run the Cisco IMC Emulator, you must first create a bridge or TAP network on the host. See the following sections for guidelines on creating a bridge or TAP interface:

- [TAP Network Mode, on page 5](#)
- [Bridge Mode with DHCP Support, on page 5](#)
- [Bridge Mode with Static IPs, on page 6](#)

TAP Network Mode

The Linux operating system enables users to create virtual network interfaces called the TAP devices. A TAP device is entirely virtual and managed by the kernel. User applications (for example, Cisco IMC Emulator) can interact with the TAP devices as if they were real. In the background, Linux pushes the packets into the regular networking stack as required, thus making it appear as if a real device is being used.

In a **Tap Network** mode, both the tap and emulator interfaces must be in the same IP subnet. For example: tap IP at 192.168.0.10 with netmask at 255.255.255.224; emulator IP at 192.168.0.11 with netmask at 255.255.255.224. In this mode, you can access the emulator only from the local host machine where it has been set up. You cannot access the emulator externally.

Bridge Mode with DHCP Support

A bridge is a layer two device that connects many Ethernet networks together to form a single larger network. A basic transparent bridge passes all packets arriving on its input port to the output ports.

In the Bridge mode, a physical interface (for example, eth0) is connected to the bridge. The packets received from emulator are forwarded to the physical interface through the bridge. Likewise, the bridge forwards the packets received from the physical interface to the emulator. This enables users to access the Cisco IMC Emulator externally.

The **Bridge Mode with DHCP Support** mode must have the following interfaces:

- A tap interface for the emulator
- A physical ethernet interface for external communication, and for fetching the IP address from the DHCP server for the emulator.

Consider a scenario where a bridge interface retrieves the IP address from a DHCP server. Assume that the IP address is 10.0.0.1. Assume too that the emulator fails to retrieve this IP address from the DHCP server. The emulator then uses the default IP address, 192.168.0.10 with netmask 255.255.255.224, from the configuration file. As the subnets are different, you cannot access the emulator. In this case, you might have to set up the bridge IP manually (using the ifconfig command) to match the same subnet of the emulator IP.



Note If you access the host machine through a remote session such as SSH or VNC, the connection may be lost for a few seconds until the bridge obtains an IP address from the DHCP server.

Bridge Mode with Static IPs

The **Bridge Mode with Static IPs** mode is similar to the **Bridge Mode with DHCP Support**. However, users must provide the IP address manually for the bridge and emulator in this mode.

Use this mode only when a DHCP server is not available in the user network. If you use this mode even when the DHCP server is available in your network, the emulator ignores the IP that you specified and obtains the IP from the DHCP server.



Note If you access the host machine through a remote session such as SSH or VNC, the connection is completely lost until the bridge IP address is set up manually. Run the script directly on the host machine.

Creating a Bridge or TAP Network through the CLI

Run the **network.sh** script to create the tap or the bridge network interface on the host machine to communicate with the emulator.

Procedure

- Step 1** At the command prompt, navigate to the extracted release package folder.
- Step 2** From the release package folder, navigate to the **prod/cxxx-m4-emu-cimc.x.x.xxxx** folder .
For example, typing **\$ cd prod/c240-m4-emu-cimc.3.0.3s18** takes you to the C240 M4 server folder.
- Step 3** Type **\$sudo ./network.sh** at the command prompt to run the **network.sh** script.
- Step 4** Configure your network mode by using any one of the following options:
 - **Tap Network**
 - **Bridge Mode with DHCP support**
 - **Bridge Mode using Static IPs**

Use the **Remove Tap Network** option or the **Remove Bridge Network** option to remove the existing network configurations.

- Step 5** Optionally, you can set a MAC address through the **Set Mac and Serial number** option. If you do not use this option, the MAC address of the emulator is automatically set.
The MAC address is a 6-byte hexadecimal number. You can only use numbers ranging from 0 to 9 and letters from A to F. The first byte should be an even number. For example, 32-D0-56-F2-B5-12 or 00-80-C8-E3-4C-BD.
 - Step 6** After configuring the network, type **Exit** to exit the script.
-

Creating a Bridge or TAP Network through the Web UI

Run the **launch.sh** script to create the tap or the bridge network interface on the host machine to communicate with the emulator.

Procedure

- Step 1** At the command prompt, type `./launch.sh` to run the `launch.sh` script.
Run the script with root privileges.
- Step 2** Go to the Emulator Configuration page by typing the following URL in a Web browser: `http://127.0.0.1:8080/`
The **IMC Emulator Configuration** page appears.
- Step 3** In the **IMC Emulator Configuration** page, expand the **Configuration** section and click the **Create/Edit Instance** option.
- Step 4** In the **Create/Edit Instance** page, enter a name for the emulator instance and select the UCS server model.
Cisco IMC Emulator supports C220 M4, C240 M4, and C460 M4 servers.
- Step 5** In the **Create/Edit Instance** page, choose the network configuration option.
Configure your network mode by using any one of the following options:
- **Tap Network**
 - **Bridge Mode with DHCP support**
 - **Bridge Mode using Static IPs**
- Note** Use the **Remove Tap Network** or the **Remove Bridge Network** option to remove the existing network configurations.
- Step 6** Optionally, you can set a MAC address through the **Set MAC Address** option. If you do not use this option, the MAC address of the emulator is automatically set.
The MAC address is a 6-byte hexadecimal number. You can only use numbers ranging from 0 to 9 and letters from A to F. The first byte should be an even number. For example, 32-D0-56-F2-B5-12 or 00-80-C8-E3-4C-BD.
- Step 7** Click **Create** to create the bridge or tap interface.
-

Viewing the Hardware Specifications of the Supported Servers

Cisco IMC Emulator supports the C220 M4, C240 M4, and C460 M4 servers. You can view the hardware specifications (CPU, DIMM, and PSU configurations) of the supported C-Series servers.

Procedure

- Step 1** Go to the Emulator Configuration page by typing the following URL in a Web browser: `http://127.0.0.1:8080/`
The **IMC Emulator Configuration** page appears.
- Step 2** In the **IMC Emulator Configuration** page, expand the **Configuration** section and click the **Templates** option.
- Step 3** In the **Templates** page, select the UCS server model.
The hardware specifications of the selected server appear.
-

Using Cisco IMC Emulator

Starting Cisco IMC Emulator through the CLI

Before You Begin

Configure the Bridge or TAP network first before performing this procedure.

Procedure

- Step 1** At the command prompt, type the server-specific script file and the tap network name. For example, type `./UCSC-C240-m4_cimc.sh tap0` to launch the emulator instance of a C240 M4 server. Here, tap 0 is the network configured previously using the `network.sh` script.
- Step 2** Wait for 10 to 15 minutes for the emulator to start the BMC services. A message appears after the emulator starts successfully. BMC is configured with `<IP address>`. IP Emulator is launched successfully. In **Bridge Mode with Static IPs** and the **Tap Network** mode, the IP address is specified by the user when creating the network. In **Bridge Mode with DHCP Support**, the emulator IP is dynamic as it is fetched from the DHCP server.
- Step 3** Use the IP address to access the Web UI, CLI, and XML API of the server.
- **Web User Interface**—Enter the following information in a web browser: `https://<IP address configured for the emulator>` .
 - **XMI API Interface**—Enter the following information in a web browser: `https://<IP address configured for the emulator>/visore.html`
 - **CLI User Interface**—Enter the following in the command prompt: `ssh admin@<IP address configured for the emulator>`
- Step 4** Use the following default credentials to access the emulator:
- User name—admin
 - Password—password
- Step 5** To exit the emulator, press **Control+C** at the command prompt.
-

Starting Cisco IMC Emulator through the Web UI

Before You Begin

Create the Bridge or TAP interface first before performing this procedure.

Procedure

- Step 1** Go to the Emulator Configuration page by typing the following URL in a Web browser: `http://127.0.0.1:8080/`

The IMC Emulator Configuration page appears.

Step 2 In the **IMC Emulator Configuration** page, click the **Rack Servers** section.
The following options appear:

- **Name:** The name of the emulator instance.
- **Model:** The server model type.
- **Serial:** The serial number of the emulator.
- **Status:** The current status of the emulator, whether running or stopped.
- **Connect:** The access options. You can access Web UI, XML API, and CLI interface of the emulator using the **Connect** option.
- **Edit Sensors:** The option to edit the sensor values.
- **Remove:** The option to remove the emulator instance and its associated network configuration.
- **Log:** Information about the current status of the emulator.

Step 3 Click **Start** in the **Start/Stop EMU** column.
Wait for the emulator to start.

Step 4 After the emulator starts, click **Connect** and select from one of the following interface options:

- **WebUI**
- **CLI**
- **XML**

Step 5 Use the following default credentials to access the emulator:

- User name—admin
- Password—password

Step 6 To stop the emulator, click **Stop** in the **Start/Stop EMU** column.

Editing Sensor Values through the CLI

The configuration file (UCSC-CXXX-M4-Config.ini) is a part of the release package. Use the configuration file to edit the sensor values to simulate a desired behavior.

You can update the configuration file even when the emulator is running. The emulator reads the values dynamically and updates them accordingly.

The following table lists some of the configuration parameters available in the configuration file.

Table 1: Cisco IMC Emulator Configuration File

//PSU1 configurations	<ul style="list-style-type: none">• PSU1_VOUT = 12• PSU1_IOUT = 9• PSU1_VIN = 10• PSU1_IIN = 10• PSU1_TEMPERATURE = 50
//PSU2 configurations	<ul style="list-style-type: none">• PSU2_VOUT = 12• PSU2_IOUT = 9• PSU2_VIN = 11• PSU2_IIN = 11• PSU2_TEMPERATURE = 55
//Temperature Sensor values	<ul style="list-style-type: none">• RISER1_INLET_TMP = 50• RISER2_INLET_TMP = 65• RISER1_OUTLET_TMP = 40• RISER2_OUTLET_TMP = 39• RISER1_SLOT1_TMP = 30

Procedure

-
- Step 1** In the release package folder, browse to the location that contains the configuration file and open the file. Alternatively, at the command prompt, type the edit command to open the configuration file. For example, typing **\$ gedit UCSC-C240-M4-Config.ini** opens the UCS C240 M4 server configuration file.
- Step 2** In the configuration file, modify the sensor values and save the file.
-

Editing Sensor Values through the Web UI

You can edit the sensor values in the configuration file to simulate a desired behavior. You can update the configuration file even when the emulator is running. The emulator reads the values dynamically and updates them accordingly.

Procedure

- Step 1** Go to the Emulator Configuration page by typing the following URL in a Web browser: `http://127.0.0.1:8080/`
The IMC Emulator Configuration page appears.
- Step 2** In the **IMC Emulator Configuration** page, click the **Rack Servers** section.
- Step 3** Click **Edit** in the **Edit Sensors** column.
The **Sensor Configuration** page appears.
- Step 4** In the **Sensor Configuration** page, edit the sensor values and click **Save**.
To reset the sensors to default values, click **Reset to Default**.
-

Troubleshooting

Problem Installing tunctl package in Linux

Issue: The following error message appears when running the `$yum install tunctl` command with root privileges on the Linux host:
no tunctl package available

Solution: If this error appears, download the package as rpm and install the package using the following command:

```
$ rpm -i tunctl.x.x.x.x.rpm
```

Problem Accessing the Emulator Web UI

Issue: Unable to access the emulator through the web UI.

Solution:

- Make sure the Linux host is set to the current date and time.
- Make sure your browser supports the TLS V1.2 protocol. For a list of browsers that support the TLS V1.2 protocol, see https://en.wikipedia.org/wiki/Transport_Layer_Security#Web_browsers
- If your browser version doesn't support the TLS V1.2 protocol, then change the following settings:

```
ssh admin@IP address of emulator #ssh admin@192.168.0.1
scope http
set http-redirect no
commit
```

Now use the address, `http://192.168.0.1` to launch the emulator web UI. This method establishes an unsecured connection to the emulator.

Problem Accessing Emulator Web UI Using Firefox 45.4.0

Issue: Unable to access the emulator through the web UI when using Firefox 45.4.0.

Solution:

More than one SSL certificate entries might have been saved in the Firefox browser. Delete the SSL certificate entry from the browser cache.

To delete the SSL certificate entries, do the following:

- 1 Browse to **Settings > Preferences > Advanced > Certificates**.
- 2 In the **Certificates** tab, click **View Certificates**.
- 3 In the **Certificate Manager** dialog box, click the **Servers** tab.
- 4 In the **Servers** tab, delete the unknown server SSL certificates.
- 5 After deleting the unknown server SSL certificates, click the **Authorities** tab.
- 6 In the **Authorities** tab, delete the Cisco self-signed certificates.

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: <http://www.cisco.com/go/trademarks>. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

© 2017 Cisco Systems, Inc. All rights reserved.



Americas Headquarters
Cisco Systems, Inc.
San Jose, CA 95134-1706
USA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV
Amsterdam, The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at www.cisco.com/go/offices.