CIM Provider Package 3.4

Version 1.0

Installation Guide

Emulex, 3333 Susan Street
Costa Mesa, CA 92626

Supported Provider Versions

UCNA Providers					
1.4					
FC Providers					
2.4					

Table of Contents

Introduction

The Emulex CIM Provider enables comprehensive management of Emulex host bus adapters (HBAs) and universal converged network adapters (UCNAs). It uses an industry standard API Common Manageability Programming Interface (CMPI v2.0) to manage various Emulex adapters.

During installation, the Emulex CIM Provider registers with a Web-based Enterprise Management (WBEM) server running a CIM Object Manager (CIMOM) service. It then internally communicates with the Emulex management API/drivers and handles inquiries and requests from various CIM clients.

Emulex provides the following two CIM Providers:

1. Emulex UCNA CIM Provider– supports the basic UCNA (FCoE, iSCSI, and NIC) modelling. This provider supports the inventory and active management of UCNAs.
2. Emulex FC HBA CIM Provider– supports the basic inventory and active management of FC cards. This provider also supports asynchronous event generation and delivery using the Indication profile.

Supported Platforms

Table 1 shows the platforms supported with the Emulex CIM Provider Kits.

Table 1. Emulex CIM Provider Supported Platforms

Operating Systems	Emulex CIM Provider Kits
VMware ESX COS 3.5	emulex_fc_provider-<*kit version*>-rhel3ASEsx35cos.i386.rpm emulex_ucna_provider-<*kit version*>-rhel3ASEsx35cos.i386.rpm
VMware ESX COS 4.0 and 4.1	emulex_fc_provider-<*kit version*>-rhel5ServerEsx40cos.x86_64.rpm For IBM Platform Agent use the RHEL 5 kit For inbox sfcb CIMOM use the OCM Core Kits
VMware ESX 5.0	VMW-ESX-5.0.0-emulex-cim-provider-<*kit version*>-<*offline bundle.zip*> For IBM Platform Agent use the RHEL 5 kit For inbox sfcb CIMOM use the OCM Core Kits
VMware Visor 4.0 and 4.1	vmware-esx-provider-emulex-cim-provider-400.<*kit version*>.zip vmware-esx-provider-emulex-cim-provider-410.<*kit version*>.zip
SUSE Linux Enterprise Server (SLES) 9	emulex_fc_provider-<*kit version*>-sles9.i586.rpm emulex_fc_provider-*kit version*>-sles9.x86_64.rpm
SLES10	emulex_fc_provider-<*kit version*>-sles10.x86_64.rpm emulex_fc_provider-<*kit version*>-sles10.i586.rpm emulex_ucna_provider-<*kit version*>-sles10.i586.rpm emulex_ucna_provider-<*kit version*>-sles10.x86_64.rpm
SLES11	emulex_fc_provider-<*kit version*>-sles11.i586.rpm emulex_ucna_provider-<*kit version*>-sles11.i586.rpm emulex_fc_provider-<*kit version*>-sles11.x86_64.rpm emulex_ucna_provider-<*kit version*>-sles11.x86_64.rpm
RHEL 4	emulex_fc_provider-<*kit version*>-rhel4.x86_64.rpm emulex_fc_provider-<*kit version*>-rhel4.i386.rpm
RHEL 5	emulex_ucna_provider-<*kit version*>-rhel5.i386.rpm emulex_fc_provider-<*kit version*>-rhel5.i386.rpm emulex_ucna_provider-<*kit version*>-rhel5.x86_64.rpm emulex_fc_provider-<*kit version*>-rhel5.x86_64.rpm

Operating Systems	Emulex CIM Provider Kits
RHEL 6 64 bit KVM	emulex_fc_provider_pegasus-<*kit version*>-rhel6.x86_64.rpm emulex_ucna_provider_pegasus-<*kit version*>-rhel6.x86_64.rpm **Note:** Pegasus rpms are released for RHEL6 64 bit KVM.
Windows Pegasus compatible	emulex_fc_provider_pegasus-<*kit version*>.exe emulex_fc_provider_pegasus_x64-<*kit version*>.exe emulex_ucna_provider_pegasus-<*kit version*>.exe emulex_ucna_provider_pegasus_x64-<*kit version*>.exe
Windows IBM Director compatible	emulex_fc_provider.msi emulex_fc_provider_x64.msi emulex_ucna_provider.msi emulex_ucna_provider_x64.msi

Supported CIM Provider Profiles

Profiles Supported for both Emulex UCNAs and FC HBAs

- Profile Registration - DMTF DSP1033
- Software Inventory - DMTF DSP1023
- Host Discovered Resources - SNIA SMI-S 1.5 Part 6 Clause 7
- FCoE Initiator Ports - SNIA SMI-S 1.5 Part 2 Clause 22
- Software Update - DMTF DSP1025
- Record Log - DMTF DSP1010
- Access Points - SNIA SMI-S 1.5 Part 2 Clause 23
- Physical Package - SNIA SMI-S 1.5 Part 2 Clause 31
- PCI Device - DMTF DSP1075
- Physical Assets Profile - DMTF DSP1011
- Storage HBA - SNIA SMI-S 1.5 Part 6 Clause 6
- Indication Profile (partial) - DMTF DSP1054

Profiles Supported for Emulex UCNAs Only

- Ethernet Port - DMTF DSP1014
- Host LAN Network Port - DMTF DSP1035
- IP Interface - DMTF DSP1036
- Diagnostics (CDMv2) - DMTF DSP1002
- iSCSI Initiator - SNIA SMI-S 1.5 Part 6 Clause 9
- iSCSI Initiator Ports Profile - SNIA SMI-S 1.5 Part 2 Clause 16

Profiles Supported for Emulex FC HBAs Only

- FC HBA - SNIA SMI-S 1.5 Part 6 Clause 5
- FC Initiator Ports Profile - SNIA SMI-S 1.5 Part 2 Clause 17

Installing the Emulex CIM Provider

There are two ways to install the Emulex CIM Provider:

1. In Linux, use the standard RPM install commands.

   -Or-

   In Windows, run the installation exe/msi.
2. Manual installation.

Prerequisites

- An Emulex OneConnect UCNA for a UCNA provider
- An FC HBA for an FC provider
- The appropriate adapter driver
- The corresponding Emulex OneCommand Manager Core Kit

Note: If you do not know the driver version and corresponding OneCommand Manager Core Kit that must be installed on your system, contact an authorized Emulex technical representative.

Note: The UCNA CIM Provider kits are tested with OneCommand Manager Core Kit version 5.1 only, on all non-Windows platforms.

Manually Installing the Emulex CIM Provider with an IBM Platform Agent

In Linux/VMware COS

To manually install the Emulex CIM Provider:

1. Extract the Emulex CIM Provider RPM.
2. Place the libraries/binaries in the directory as required by the Open Pegasus CIMOM.

   This path is configurable and depends on the customization performed in the CIMOM.

   For example:

   In tog-pegasus CIMOM, the path is /usr/lib/Pegasus/providers/.

   In the development build of Open Pegasus, the path is $PEGASUS_HOME/bin.

   Refer to the CIMOM release notes for the exact path.

   In Linux and VMware COS, the extracted libraries/binaries are libelxhba_cmpi_provider.so/ libemulex_ucna_provider.so.
3. Ensure that there is no "ldd" dependency for the extracted libraries/binaries.
4. Compile the extracted mof files. Ensure that the $PEGASUS_HOME variable is set in the shell and the "$PEGASUS_HOME/bin" is included in the system path ($PATH) variable. Execute the following commands from the same shell:
   - `cimmofl -aEV -n root/emulex elxhba_schema.mof`

     This command compiles the Emulex CIM Provider schema file.
   - `cimmofl -aEV -n root/pg_interop elxhba_interop_schema.mof`

     This command compiles the Emulex CIM Provider schema file for SLP registration.

- `cimmofl -aEV -n root/pg_interop elxhba_reg.mof`
  This command compiles the Emulex CIM Provider registration file.

5. Restart the CIMOM.
6. Check the installation using the following command:
   `cimprovider -ls`
   The status of "elxhba_cmpi_provider_Module/emulex_ucna_provider_Module" must be "OK".

In VMware Visor ESXi 4.0/4.1

To install the Emulex CIM Provider in a VMware Visor environment, use the esxupdate command line utility and perform the following steps:

1. Enable SSH on the VMware Visor host.
2. Transfer the Emulex CIM Provider vib file to the VMware Visor host.
3. Login to the VMware Visor host and execute the following command:
   `esxupdate -b file://Emulex Provider vibfilepath --nosigcheck --maintenancemode update`

In ESX 5.0

To install the Emulex CIM Provider using offline bundle:

`esxcli software vib install -d=<offline-bundle.zip>`

Repairing a Faulty Emulex CIM Provider Installation in Linux

Uninstalling Open Pegasus

To uninstall an existing version of Open Pegasus:

1. Query for the installed Pegasus packages.

   ```
   rpm -ql `rpm -qa|grep -i peg`
   ```

2. Remove the Pegasus packages.

   ```
   rpm -e `rpm -qa|grep -i peg`
   ```

3. Manually delete directories, if any, created by Pegasus.

Checking for Other CIMOMs

To check if any other CIMOM is running:

> **Note:** It is important to check for multiple CIMOMs because on some operating systems other CIMOMs exist as inbox (OpenWbem on SLES 10 and sfcb on SLES 11).

1. Get the CIMOM process.

   ```
   ps -ef|grep -I cim
   ```

2. Identify the CIMOM installed.

   ```
   which <CIMOM process>
   ```

3. Stop the corresponding CIMOM services.

   ```
   /etc/init.d/<service name> stop
   ```

4. Ensure that no other CIM-related process is running by performing Step 1 again.
5. In case any CIMOM process is still running, kill the hanging processes.

   ```
   killall -9 <CIMOM process>
   ```

Reinstalling Open Pegasus

To reinstall Open Pegasus:

1. Install the Open Pegasus RPM.
2. Restart the CIMOM.

   `/etc/init.d/tog-pegasus restart` (or equivalent command)

3. Check the health of the newly installed CIMOM.
   - `cimprovider -ls`
   - `cimconfig -lc`
4. Use the "osinfo" application to see if the Open Pegasus is able to cater to CIM requests.
5. In case it is required, update the file /etc/Pegasus/access.conf to correct the permissions. Restart the CIMOM for settings to take effect.
6. Query and check for provider response using either the cimcli or wbemcli CIM client. Query the root/pg_interop namespace to verify that Open Pegasus is working as expected.

7. Note the <PEGASUS_HOME> variable for this distribution of Open Pegasus. This is usually /var/lib/Pegasus.
8. Use the following command and note the "providerDir" variable for this distribution of Open Pegasus.

   `cimconfig -lc`

Note: This path may be relative to <*PEGASUS_HOME*>.

Reinstalling the Emulex CIM Provider

To reinstall the Emulex CIM Provider:

1. Copy the Emulex Provider binaries to the "providerDir" noted in Step 8 of “Reinstalling Open Pegasus” on page 5.
2. Register the Emulex CIM Provider using the following commands:
   - `cimmofl -aEV -n root/emulex <provider name>_schema.mof -R <PEGASUS_HOME>`
   - `cimmofl -aEV -n root/pg_interop <provider name>_interop_schema.mof -R <PEGASUS_HOME>`
   - `cimmofl -aEV -n root/pg_interop <provider name>_reg.mof -R <PEGASUS_HOME>`

   <*PEGASUS_HOME*> represents the variable you noted in Step 7 of “Reinstalling Open Pegasus” on page 5.
3. Restart the CIMOM.

   `/etc/init.d/tog-pegasus restart` (or equivalent command)
4. Check the health of the CIMOM using steps detailed in “Reinstalling Open Pegasus” on page 5.
5. Enumerate the root/emulex namespace.

   In case of errors, perform the steps detailed in the following section “Generating Provider Logs” on page 6.

Generating Provider Logs

To enable provider logs:

1. Enumerate the ELXHBA_RecordLog (for an FC HBA) or the ELXUCNA_RecordLog (for a UCNA) class and note the provider log instance. If enumeration fails, perform the steps detailed in “Generating Provider Logs When Enumeration Fails” on page 7.
2. Execute the command:

   `wbemcli -noverify cm 'https://root:<password>@<IP>/root/emulex:<Instance from step 1> ' SetLogParams.LogLevel=5,logmode=2,tracepath='<file path>'`
3. Execute the following command:

   `wbemcli -noverify cm 'https://root:<password>@<IP>/root/emulex:<Instance from step 1> ' RequestStateChange.RequestedState=2`
4. Perform the operation that is not working as expected. The provider logs are available in the <*file path*> specified in Step 2.

5. Contact an Emulex technical support representative with the provider logs and the system logs (/var/log/messages, or /var/log/syslog.log in case of ESX 5.0).

To disable provider logs:

1. Execute the following command:

   `wbemcli -noverify cm 'https://root:<password>@<IP>/root/emulex:<Instance from step 1> ' RequestStateChange.RequestedState=3`

Generating Provider Logs When Enumeration Fails

To enable provider logs when enumeration fails:

1. Stop the CIMOM.
2. Create a file "emulex_fc_provider.dmp (for FC HBA)" or "emulex_ucna_provider.dmp (for UCNA)" in "/etc" location, and manually enter the following two lines, without spaces:

   ```
   2,5,2
   /tmp/providerlogs.txt
   ```

3. Start the CIMOM and enumerate the Emulex classes in the "root/emulex" namespace.
4. Perform the operation which is not working as expected. The provider logs are available in </tmp/providerlogs.txt> specified in Step 2.
5. Contact an Emulex technical support representative with the provider logs and the system logs (/var/log/messages, or /var/log/syslog.log in case of ESX 5.0).

Additional Information

After you have performed all the steps detailed to repair a faulty CIM Provider installation, contact an Emulex Technical representative with the following additional information:

In ESXi (VMware Visor environment)

Output from the following commands:

- `esxcfg-scsidevs -a`

ESX 5.0

- `esxcli software vib list | grep -i be2`
- `esxcli software vib list | grep -i lpfc`
- `esxcli software vib list | grep -i emu`

ESX 4.0/4.1 COS/VISOR

- `esxupdate –vib-view query | grep -i be2`
- `esxupdate –vib-view query | grep -i lpfc`
- `esxupdate –vib-view query | grep -i emu`

In Linux

1. Output of any one of the following commands:

   `/usr/sbin/hbanyware/hbacmd listhbas`

   Or
   `/usr/sbin/ocmanager/hbacmd listhbas`

2. Restart the CIMOM and then enumerate the ELXUCNA_SoftwareIdentity and ELXHBA_SoftwareIdentity classes in the "root/emulex" namespace.

   For example:
   `wbemcli -nl -noverify ein 'https://root:<password>@<IP address>/root/emulex:ELXHBA_SoftwareIdentity'`

3. Provide the system logs /var/log/messages, or /var/log/syslog.log in case of ESX 5.0) for the operation performed in Step 2.
4. Output from the following commands:
   - `rpm -qa| grep -i elx`
   - `rpm -qa| grep -i lpfc`
   - `rpm -qa| grep -i be2`
   - `lspci`

Frequently Asked Questions (FAQs)

Here you will find answers to questions that you may encounter while using the Emulex CIM Provider.

If you still have questions that have not been answered in the FAQs or require additional information, contact an authorized Emulex technical support representative at tech.support@emulex.com, 800-854-7112 (US/Canada toll free), +1714-885-3402 (US/International), or +44 1189-772929 (Europe, Middle East, and Africa).

How to Update the Firmware on an Emulex Card?

The Software Update profile is supported on both the Emulex UCNA CIM Provider and the Emulex FC HBA CIM Provider. You must use this profile to update the firmware on the Emulex cards. The following methods are implemented in the Emulex specific Software Update profile:

1. InstallfromByteStream – requires a custom CIM client, which can read the firmware file and create a ByteStream that is used for updating the firmware.
2. InstallfromURI – supports two different types of URI (Uniform Resource Identifiers).
    a. The firmware file to be upgraded is available locally on the machine hosting the Emulex card.
    b. The firmware file to be upgraded is available on a remote machine like an http or an https server. In this case the Emulex UCNA CIM Provider uses the “libcurl” library available on the host machine, where the CIM Provider is running, to download the firmware file.

Emulex CIM_SoftwareInstallationServiceCapabilities class has the attribute SupportedURISchemes that indicates the supported URI schemes.

To update the firmware:

1. Enumerate the CIM_SoftwareInstallationService in Emulex's namespace. Select the CIM_SoftwareInstallationService specific to the card on which the firmware is to be updated.

    For example: If the UNCA card needs a firmware update, select the ELXUCNA_SoftwareInstallationService class instance.

    ```
    wbemcli  -noverify ein 'https://root:<password>@<IP>/root/
    emulex:elxucna_softwareinstallationservice' -nl
    ```

2. Enumerate the CIM_FCPort or CIM_EthernetPort depending on the card on which the firmware is to be updated. Select the desired port from the enumerated instances.

    For example:

    ```
    wbemcli  -noverify ein 'https://root:<password>@<IP>/root/
    emulex:elxucna_ethernetport' -nl
    ```

3. Select the correct firmware file and choose one of the two methods detailed in Step 1 and 2 to update the firmware.

    For example:

    ```
    wbemcli -nl -noverify cm 'https://root:<password>@<IP>/root/
    emulex:<elxucna_softwareinstallationservice instance>'
    InstallFromURI.URI="<Firmware file path>",Target=<elxucna_ethernetport instance>
    ```

How to Enable Logs or Collect Symptoms?

The Record Log profile is supported on both the Emulex UCNA CIM Provider and the Emulex FC HBA CIM Provider.

To enable logs:

1. Enumerate the specific instance of the CIM_RecordLog class. For example, if the traces need to be enabled for an UCNA CIM Provider, enumerate the ELXUCNA_RecordLog instance.
2. Invoke the method SetLogParams with “loglevel=5”, “logmode=2”, and “tracepath=tracefile path”.
3. Invoke the method RequestStateChange with “RequestedState=2”.

   Traces are collected in the file specified in the “tracepath” attribute in Step 2.

Important checkpoints or critical errors are also logged in the /var/log/message (/var/log/syslog.log in case of ESX 5.0) file.

How to Uninstall a Corrupted/Bad OneCommand Manager Core Kit?

Use the “-nodeps” option to uninstall an existing corrupted/bad version of the OneCommand Manager Core Kit.

What is the Supported CIMPLE Version?

CIMPLE version 2.0.14 is supported for both the Emulex UCNA CIM Provider and the Emulex FC HBA CIM Provider.

How to Extract Windows Binaries in MSI Packages?

For MSI packages, use the “msiexec” command to extract the files without installing the package.

To extract Windows binaries:

1. Click **Start**> **Run** and open the command prompt.
2. Run the following command:

   `msiexec /a <msi file>`

3. Specify the path where the files must be extracted. Follow the wizard to complete the process.

   The files are available in the specified file location.

   For *.exe packages, use the following command:
   `<*.exe> /install:<local path>`

For example: The command “c:\>emulex_fc_provider_pegasus_x64-2.4.10.1.exe /install:C:\temp”, extracts all the files from “emulex_fc_provider_pegasus_x64-2.4.10.1.exe” to “c:\temp” directory.

How to Execute a Diagnostic Test with the Emulex CIM Provider?

To execute a diagnostic test with the Emulex CIM Provider:

1. Get the instance of the managed element (ethernetport). For example:

   ```
   wbemcli  -noverify ein 'https://root:<password>@<IP>/root/
   emulex:elxucna_ethernetport' -nl
   ```

2. Get the instance of the ELXUCNA_DiagnosticTest class. For example:

   ```
   wbemcli  -noverify ein 'https://root:<password>@<IP>/root/
   emulex:elxucna_diagnostictest' -nl
   ```

3. Invoke the method RunDiagnosticservice on the elxucna_diagnostictest. For example:

   ```
   wbemcli -noverify cm 'https://root:<password>@<IP>/root/emulex:<Diagnostic Test
   instance from step 2>' RunDiagnosticservice.ManagedElement=<ManagedElement
   instance from step 1>'
   ```

4. A CIM_ConcreteJob instance is created for each diagnostic test run. For example:

   ```
   wbemcli  -noverify ein 'https://root:<password>@<IP>/root/
   emulex:elxucna_concretejob' -nl
   ```

5. Results of the diagnostic test runs are available in ELXUCNA_DiagnosticCompletionRecord class instances. For example:

   ```
   wbemcli  -noverify ein 'https://root:<password>@<IP>/root/
   emulex:elxucna_diagnosticcompletionrecord' -nl
   ```

6. The diagnostic logs can be cleared using the ClearLog function of ELXUCNA_DiagnosticsLog class. For example:

   ```
   wbemcli -noverify cm 'https://root:<password>@<IP>/root/emulex:<Corresponding
   Diagnostic log instance>' ClearLog
   ```

How to Discover vNICs Using the Emulex CIM Provider?

To determine vNICs using UCNA CIM Provider version 1.1:

1. Find all the CIM_EthernetPort instances whose SystemName property has a value equal to the name of the desired host. These instances may represent NIC, vNIC, iSCSI, or FCoE PCI functions.
2. From the resulting pool of CIM_EthernetPorts, find those associated to a CIM_LANEndpoint instance through an instance of CIM_DeviceSAPImplementation. The resulting list of CIM_EthernetPorts are either NIC or vNIC functions.
3. For each member in the NIC/vNIC CIM_EthernetPorts list; obtain the CIM_ControlledBy association associating the CIM_EthernetPort to an instance of CIM_PortController.
4. Any CIM_EthernetPort whose CIM_ControlledBy.Antecedent property has the same value as that of another CIM_EthernetPort's CIM_ControlledBy.Antecedent property, represents a vNIC.

To determine vNICs using UCNA CIM Provider version 1.2 or later:

1. Find all the CIM_VLANEndpoint instances whose SystemName property has a value equal to the name of the desired host. These instances are vNIC endpoints.
2. Find the CIM_EndpointIdentity instance associating the CIM_VLANEndpoint instance to a CIM_LANEndpoint instance. Follow the association to the CIM_LANEndpoint.
3. Find the instance of CIM_DeviceSAPImplementation instance that associates the CIM_LANEndpoint to an instance of CIM_EthernetPort. Follow the associations to the CIM_EthernetPort instance representing the vNIC.

To determine vNICs hosted by a UCNA in a particular host:

1. Find the CIM_Card instance for the UCNA.
2. Find the vNICs in the host as detailed earlier, based on the version of the CIM Provider.

3. For each CIM_EthernetPort representing a vNIC:
    a. Find the instance of CIM_ControlledBy associating the CIM_EthernetPort to an instance of CIM_PortController. Follow the association to the CIM_PortController instance.
    b. Find the instance of CIM_Realizes associating the CIM_PortController to an instance of CIM_PhysicalConnector. Follow the association to the CIM_PhysicalConnector instance.
    c. Find the instance of CIM_Container associating the CIM_PhysicalConnector to an instance of CIM_Card.
4. Each CIM_EthernetPort with a CIM_Container instance referencing the CIM_Card instance found in Step 1 is hosted by the targeted UCNA.

To determine vNICs operating over the same physical port:

1. Find the CIM_PhysicalConnector instance representing the port in question. The key value for CIM_PhysicalConnector instances contains the serial number of the UCNA hosting the port and the port number assigned to the physical port.
2. Find the vNICs in the host as detailed earlier, based on the version of the CIM Provider. For each CIM_EthernetPort representing a vNIC:
    a. Find the instance of CIM_ControlledBy associating the CIM_EthernetPort to an instance of CIM_PortController. Follow the association to the CIM_PortController instance.
    b. Find the instance of CIM_Realizes associating the CIM_PortController to an instance of CIM_PhysicalConnector.
3. Each CIM_EthernetPort with a CIM_Realizes instance referencing the CIM_PhysicalConnector instance found in Step 1 is running over the targeted physical port.