

HP Virtual Connect Enterprise Manager SOAP API guide



Legal notices

© Copyright 2008 Hewlett-Packard Development Company, L.P.

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Contents

Introduction.....	7
Overview of SOAP APIs	7
Overview of VCEM.....	7
Configuration	9
Configuring the CMS	9
Configuring the computer that runs the external application	9
Ports.....	9
Logging in to the CMS.....	10
Available SOAP APIs.....	11
login	11
Summary	11
API	11
Parameters.....	11
Return value	11
Exceptions	11
Minimum privileges	11
logout	11
Summary	11
API	11
Parameters.....	11
Return value	12
Exceptions	12
Minimum privileges	12
getProductVersion	12
Summary	12
API	12
Parameters.....	12
Return value	12
Exceptions	12
Minimum privileges	12
Sample code.....	12
listVCDomains	13
Summary	13
API	13
Parameters.....	13
Return value	13
Exceptions	14
Minimum privileges	14
Objects and attributes	14
Sample code.....	14
listVCDomainGroups.....	15
Summary	15
API	15
Parameters.....	15
Return value	15
Exceptions	16
Minimum privileges	16
Objects and attributes	16
Sample code.....	17
listDetailsForSpecifiedProfile.....	18
Summary	18
API	18
Parameters.....	18

Return value	18
Exceptions	18
Minimum privileges	18
Objects and attributes	18
Sample code	20
listServerProfiles	21
Summary	21
API	21
Parameters	21
Return value	21
Exceptions	21
Minimum privileges	21
Sample code	21
getVCDomainForEnclosure	22
Summary	22
API	22
Parameters	22
Return value	22
Exceptions	23
Minimum privileges	23
Sample code	23
getVCDomainGroupForEnclosure	24
Summary	24
API	24
Parameters	24
Return value	24
Exceptions	24
Minimum privileges	24
Sample code	24
isEnclosureLicensed	25
Summary	25
API	25
Parameters	25
Return value	25
Exceptions	25
Minimum privileges	25
Sample code	25
listStatusForMvcdJob	26
Summary	26
API	26
Parameters	26
Return value	26
Exceptions	27
Minimum privileges	27
Objects and attributes	27
Sample code	27
createProfile	28
Summary	28
API	28
Parameters	28
Return value	29
Exceptions	29
Minimum privileges	29
Usage notes	29
Objects and attributes:	29
Sample code	30
assignProfile	32
Summary	32
API	32
Parameters	32
Return value	32

Exceptions	32
Minimum privileges	32
Usage notes	32
Sample code	32
migrateProfile	34
Summary	34
API	34
Parameters	34
Return value	34
Exceptions	34
Minimum privileges	34
Usage notes	34
Sample code	34
checkPowerStatusOfBay	36
Summary	36
API	36
Parameters	36
Return value	36
Exceptions	36
Minimum privileges	36
Sample code	36
powerBayOn	37
Summary	37
API	37
Parameters	37
Return value	37
Exceptions	37
Minimum privileges	37
Usage notes	37
Sample code	38
powerBayOff	38
Summary	38
API	38
Parameters	39
Return value	39
Exceptions	39
Minimum privileges	39
Usage notes	39
Sample code	39
Implementation	41
WSDL	41
Fault exceptions	41
HP services and technical support	45
Where to go for additional help	45
HP contact information	46

Introduction

This guide outlines how to configure and implement Simple Object Access Protocol (SOAP) Application Program Interfaces (APIs) in HP Virtual Connect Enterprise Manager (VCEM). Using this guide, you can perform multiple tasks, such as:

- Viewing available SOAP APIs and their usage
- Viewing possible error conditions
- Configuring SOAP API functionality
- Configuring the VCEM Central Management Server (CMS) and the computer that runs the external application

Sample SOAP API code is also provided, enabling you to quickly implement this feature of VCEM.

Overview of SOAP APIs

SOAP APIs are simple tools that enable you to perform tasks that extend and customize the capabilities of VCEM. You can use SOAP APIs to retrieve and make changes to a given Virtual Connect (VC) Domain using the Virtual Connect Manager (VCM).

You can create both simple and complex tasks using SOAP APIs. For example, you can use SOAP APIs to:

- Power up a server bay
- Power up multiple server bays across multiple enclosures, produce a list of all profiles set up on those bays, and then power the bays back down

You access and manipulate these SOAP APIs with a user-provided external application (such as `testAPI.java`). Sample SOAP API code to be used with this external application is provided in this document.

NOTE: The VCEM SOAP server for the SOAP APIs is Axis 1.1.

Overview of VCEM

VCEM simplifies the management of multiple HP BladeSystem enclosures that use HP Virtual Connect to control Local Area Network (LAN) and Storage Area Network (SAN) connectivity, helping organizations increase productivity, respond more quickly to business demands, and significantly reduce operating costs.

Building on existing HP Virtual Connect technology, VCEM provides a single console to manage network infrastructure connections, a central resource pool for LAN and SAN address administration, and grouping capabilities that enable rapid server deployment and movement.

Use VCEM in environments with multiple HP BladeSystem enclosures (typically four or more) to complete critical data center tasks quickly, reliably, and with minimal disruption to the production LAN and SAN. These tasks include:

- Installing server blades
- Performing server blade recovery
- Scheduling system maintenance
- Deploying large virtual connect installations
- Rapidly migrating server blades (from test to production environments, or between data center locations) on the same network infrastructure

VCEM delivers new levels of management and control for HP BladeSystem environments that use HP Virtual Connect to maintain server, LAN, and SAN infrastructures. Future versions of VCEM will provide expanded capabilities that further automate infrastructure management and server availability.

For more information on VCEM, see <http://www.hp.com/go/vcem>.

Configuration

This chapter describes how to configure the VCEM CMS and the computer on which the external application runs. You must deploy and configure both devices before using SOAP APIs.

Configuring the CMS

You must have a certificate on the CMS to authenticate the computer used to run the external application. This certificate can be either:

- Self-signed
- From a Certification Authority used to represent the remote machine

Configuring the computer that runs the external application

To authenticate the computer when the external application establishes an SSL session, you must install the CMS certificate on this computer.

The keystore is application-specific. The external application must document the location and installation of a certificate into the keystore.

To access the certificate from the CMS:

1. Browse to `http://yourcms:280/GetCertificate` or `https://yourcms:50000/GetCertificate`.
2. Copy the base64-encoded CMS certificate to a file on the external application computer.

For Java-based external applications, you can use a trusted certificate keystore with the following properties:

- `javax.net.ssl.trustStore`—Contains the CMS certificate.
- `javax.net.ssl.trustStorePassword`—Contains the password.

To create these keystores and to import and export certificates, use the Java keytool command-line application. For more information on this application, see <http://java.sun.com/j2se/1.5.0/docs/tooldocs/solaris/keytool.html>.

Ports

Port 50001 must be available on the external application computer to use SOAP APIs. Before using SOAP APIs, verify this port is available.

NOTE: In some cases, port 50002 is also used for SOAP API functions.

Logging in to the CMS

This chapter describes the CMS login process and explains SOAP API sessions.

You can only log in to the CMS for a single session. Logging in using the login SOAP API, an HP Systems Insight Manager (HP SIM) username, and an HP SIM password creates a login token that verifies your identity as you perform various actions in VCEM. For example, the login token performs several functions such as verifying your access privileges to the CMS.

A login token is valid for the current session only. To log in again, you must begin a new session with a corresponding login token.

Your session ends either when you log out or when the session times out. When your session ends, your login token is no longer valid.

The session length is 30 minutes. However, each time you execute a SOAP API, the current session is extended. For example, if you execute a SOAP API, your session is now valid for 30 minutes total, regardless of the time remaining before you executed the SOAP API.

Available SOAP APIs

The following are some commonly used SOAP APIs.

For more information on the exceptions listed for these SOAP APIs, see the “Fault exceptions” section of this guide.

login

Summary

This SOAP API obtains a login token.

API

```
String login (String usr, String pwd)
```

Parameters

- Username
- Password

Return value

Login token

Exceptions

2803, 2806, 5010, 5011

Minimum privileges

Read only

logout

Summary

This SOAP API invalidates a login token and manually logs out.

API

```
void logout (String token)
```

Parameters

Login token

Return value

Void

Exceptions

2801, 2804, 2805, 2806, 2807, 2808

Minimum privileges

Read only

getProductVersion

Summary

This SOAP API returns a string containing the VCEM version (such as v1.0).

API

String getProductVersion (String token)

Parameters

Login token

Return value

String containing VCEM version

Exceptions

2145, 2801, 2804, 2805, 2806

Minimum privileges

Read only

Sample code

From the external application computer, perform the following steps:

1. Access `https://mycms.corp.com:50001/GetCertificate`.
2. Save the certificate as `sim.cer` in the current working directory.
3. Type the following text in the command line, followed by the password:

```
keytool -import -keystore sim.ks -file sim.cer  
password is "changeit"
```

4. Set a trusted certificate by entering the following SOAP API code in an external application (such as testAPI.java):


```
import java.net.URL;
public class GetProductVersionAPISample {
    public static void main(String [] args) throws Exception {
        System.setProperty("javax.net.ssl.trustStore", "sim.ks");
        System.setProperty("javax.net.ssl.trustStorePassword", "changeit");
```
5. Create a service.


```
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator service = new
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator();
```
6. Create a stub using the service.


```
String url =
"https://mycms.corp.com:50001/mxsoap/services/mvcd1_0";
com.hp.mvcd.api.v1_0soap.SoapApi vcemAPI = service.getMvcd1_0(new
URL(url));
```
7. Execute the SOAP APIs.
 - a. Log in to the CMS:


```
String token = vcemAPI.login("username","passwd");
System.out.println(vcemVersion);
```
 - b. Log out from the CMS:


```
vcemAPI.logout(token);
    }
}
```

listVCDomains

Summary

This SOAP API returns all VC Domains recognized by VCEM as a list of VC Domain objects.

Certain values are not returned by this SOAP API in real time. Because no volatile information is stored in the database, you must request the information in real time through the VC Domain. For the bay, the transient information includes all data from the server.

To retrieve the physical power state, use the CheckPowerStatusofBay () SOAP API.

API

```
VCDomain [ ] listVCDomains (String token)
```

Parameters

Login token

Return value

Multiple MVCD VC Domain objects

Exceptions

2012, 2116, 2138, 2139, 2142, 2401, 2801, 2804, 2805, 2806

Minimum privileges

Read only

Objects and attributes

VC Domain object:

- String: Name
An associated unique name for the VC Domain
- Long: vcDomainId
- String: vcDomainGUID
- String: Description
- String: VCDGName
- Long: VCDGId
- Array of associated Enclosures objects.
 - Enclosure[]: enclosures
 - String: name
 - String: description
 - String: uuid
 - Bay[]: bay
 - String: serialNumber
- String: ipAddress
 - *nnn.nnn.nnn.nnn*
- String: status
 - NOT_LICENSED("NOT_LICENSED"),
 - EXPIRED_LICENSE("EXPIRED_LICENSE"),
 - LICENSED_UNMANAGED("LICENSED_UNMANAGED"),
 - AUTHORIZED("AUTHORIZED"),
 - MANAGED("MANAGED"),
 - CRITICAL("CRITICAL"),
 - LOCK_BROKEN("LOCK_BROKEN"),
 - OUT_OF_DATE("OUT_OF_DATE");

Sample code

From the external application computer, perform the following steps:

1. Access `https://mycms.corp.com:50001/GetCertificate`.
2. Save the certificate as `sim.cer` in the current working directory.
3. Type the following text in the command line, followed by the password:

```
keytool -import -keystore sim.ks -file sim.cer  
password is "changeit"
```

4. Set a trusted certificate by entering the following SOAP API code in an external application (such as testAPI.java):


```
import java.net.URL;
public class ListVCDomainsAPISample {
    public static void main(String [] args) throws Exception {
        System.setProperty("javax.net.ssl.trustStore", "sim.ks");
        System.setProperty("javax.net.ssl.trustStorePassword", "changeit");
```
5. Create a service.


```
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator service = new
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator();
```
6. Create a stub using the service.


```
String url =
"https://mycms.corp.com:50001/mxsoap/services/mvcd1_0";
com.hp.mvcd.api.v1_0soap.SoapApi vcemAPI = service.getMvcd1_0(new
URL(url));
```
7. Execute the SOAP APIs.
 - a. Log in to the CMS:


```
String token = vcemAPI.login("username","passwd");
```
 - b. Return a list of VC Domains:


```
com.hp.mvcd.api.v1_0soap.VCDomain[] vcdList = vcemAPI.listVCDomains
(token);
for (int i = 0; i < vcdList.length; i++)
{System.out.println(vcdList[i].getName() + "\n");
}
```
 - c. Log out from the CMS:


```
vcemAPI.logout(token);
}
```

listVCDomainGroups

Summary

This SOAP API returns VC Domain Groups (as a list of VC Domain Group objects) and the basic information and status for the VC Domain Groups. This SOAP API also displays which VC Domain Groups are being updated and lists VC Domains in each VC Domain Group.

API

```
VCDomainGroup [ ] listVCDomainGroups (String token)
```

Parameters

Login token

Return value

Multiple MVCD VC Domain Group objects

Exceptions

2401, 2801, 2804, 2805, 2806

Minimum privileges

Read only

Objects and attributes

VC Domain Group object:

- String: `Name`
- String: `Description`
- Long: `vcDomainGroupId`
- String: `Status`
 - `NORMAL`
 - `INFORMATIONAL`
- VCDomain: `masterDomain`

- VCDomain[]: `domains`

Array of VC domains:

- Each “domain” object consists of information returned by the `listVCDomains()` SOAP API

- String: `macRangeType`
 - `USER-DEFINED`
 - `VC-DEFINED`
 - `HW-DEFAULTS`
 - `EXTERNALLY_DEFINED`
 - `UNKNOWN`

- String: `wwnRangeType`
 - `USER-DEFINED`
 - `VC-DEFINED`
 - `HW-DEFAULTS`
 - `EXTERNALLY_DEFINED`
 - `UNKNOWN`

- ServerProfile[]: `serverProfiles`

Array of Serverprofile objects:

- String: `name`
- String: `description`
- Long: `id`
- String: `VCDGName`
- Bay: `bay`
- NetworkConnection[]: `networkConnection`
- FCConnection[]: `fcConnection;`

- FCFabric[]: `fcFabrics`

Array of FcFabrics, each FcFabrics object:

- String: `name`

- String: description
- String: ulmData
- Integer: slot
- Network[]: networks
 - Array of Networks objects:
 - String: name
 - String: description
 - Boolean: enabled
 - String: ulmData
 - Boolean: smartLink
 - Integer: vlanId
 - Boolean: nativeNetwork
 - String: portSetMode

Sample code

From the external application computer, perform the following steps:

1. Access `https://mycms.corp.com:50001/GetCertificate`.
2. Save the certificate as `sim.cer` in the current working directory.
3. Type the following text in the command line, followed by the password:


```
keytool -import -keystore sim.ks -file sim.cer
password is "changeit"
```
4. Set a trusted certificate by entering the following SOAP API code in an external application (such as `testAPI.java`):


```
import java.net.URL;
public class ListVCDomainGroupsAPISample {
    public static void main(String[] args) throws Exception {
        System.setProperty("javax.net.ssl.trustStore", "sim.ks");
        System.setProperty("javax.net.ssl.trustStorePassword", "changeit");
```
5. Create a service.


```
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator service = new
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator();
```
6. Create a stub using the service.


```
String url =
"https://mycms.corp.com:50001/mxsoap/services/mvcd1_0";
com.hp.mvcd.api.v1_0soap.SoapApi vcemAPI = service.getMvcd1_0(new
URL(url));
```
7. Execute the SOAP APIs.
 - a. Log in to the CMS:


```
String token = vcemAPI.login("username", "passwd");
```
 - b. Return a list of VC Domain groups:


```
com.hp.mvcd.api.v1_0soap.VCDomainGroup[] vcdgList = vcemAPI
.listVCDomainGroups(token);
for (int i = 0; i < vcdgList.length; i++)
{System.out.println(vcdgList[i].getName() + "\n");
}
```

```
c. Log out from the CMS:
    vcemAPI.logout ( token );
    }
}
```

listDetailsForSpecifiedProfile

Summary

This SOAP API returns profile information for a given bay.

Values in the physicalPowerState, physicalUID, physicalProductName, physicalSerialNumber, physicalServerName, and physicalVirtualSerialNumber fields are not returned by this SOAP API in real time. Because no volatile information is stored in the database, you must request the information in real time through the VCM. For the bay, the transient information includes all data from the server.

To retrieve the physical power state, use the CheckPowerStatusofBay () SOAP API.

API

```
ServerProfile listDetailsForSpecifiedProfile (String token, String enclosureName, int bayNumber)
```

Parameters

- Login token
- Enclosure name
- Bay number

Return value

An array of server profiles

Exceptions

2401, 2801, 2804, 2805, 2806, 5012, 5013, 5045

Minimum privileges

Read only

Objects and attributes

Server Profile object:

- Long: id
- String: name
- String: description
- String: VCDGname
- Long: VCDGId

- Bay: bay
 - Long: bayId
 - Int: bayNumber
 - String: physicalPowerState
 - String: physicalUID
 - ON
 - OFF
 - NOT-APPLICABLE
 - String: physicalProductName
 - String: physicalSerialNumber
 - String: physicalServerName
 - String: physicalVirtualSerialNumber
 - String: status
 - AVAILABLE
 - PROFILE_ASSIGNED
 - Enclosure: enclosure
 - String: name
 - String: description
 - String: uuid
 - String: serialNumber
- Array of NetworkConnection, each NetworkConnection object:
 - MACAddress[]: macAddresses
 - String: macAddress
 - String: status: FREE, IN_USE, EXCLUDED, EXTERNAL
 - String: pxeBootSetting
 - DISABLED
 - ENABLED
 - USE-BIOS
 - portNumber
 - Network: network
 - String: name
 - String: description
 - Boolean: enabled
 - String: ulmData
 - Boolean: smartLink
 - Integer: vlanId
 - Boolean: nativeNetwork
 - String: portSetMode
- Array of FCConnection, each FCConnection object:
 - String: fcBootPriority
 - DISABLED: SAN boot will not be attempted.
 - PRIMARY: This port will be used as a first attempt at a SAN boot. Only one port may be primary.

- SECONDARY: This port will be used as a second attempt at a SAN boot.
 - USE_BIOS: SAN boot for this port will be used as configured by BIOS.
- String: `bootportWWN`
- Integer: `bootlun`
- Integer: `portNumber`
- WWN[]: `wwnAddress`
 - String: `wwnAddress`
 - String: `status: FREE, IN_USE, EXCLUDED, EXTERNAL`
- String: `portSpeed`
 - DISABLED: Only Disable this port
 - AUTO: Allow port to automatically negotiate the highest speed
 - 2Gb: Hard select to 2Gb
 - 4Gb: Hard select to 4Gb
- FCFabric: `fcFabric`
 - String: `name`
 - String: `description`
 - String: `ulmData`
 - Integer: `slot`

Sample code

From the external application computer, perform the following steps:

1. Access `https://mycms.corp.com:50001/GetCertificate`.
2. Save the certificate as `sim.cer` in the current working directory.
3. Type the following text in the command line, followed by the password:


```
keytool -import -keystore sim.ks -file sim.cer
password is "changeit"
```
4. Set a trusted certificate by entering the following SOAP API code in an external application (such as `testAPI.java`):


```
import java.net.URL;
public class ListDetailsForSpecifiedProfileAPISample {
    public static void main(String[] args) throws Exception {
        System.setProperty("javax.net.ssl.trustStore", "sim.ks");
        System.setProperty("javax.net.ssl.trustStorePassword", "changeit");
```
5. Create a service.


```
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator service = new
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator();
```
6. Create a stub using the service.


```
String url =
"https://mycms.corp.com:50001/mxsoap/services/mvcd1_0";
com.hp.mvcd.api.v1_0soap.SoapApi vcemAPI = service.getMvcd1_0(new
URL(url));
```
7. Execute the SOAP APIs.
 - a. Log in to the CMS:


```
String token = vcemAPI.login("username","passwd");
```

- b. Return the details of the profile assigned to the enclosure:

```
com.hp.mvcd.api.v1_0soap.ServerProfile profileObj =
vcemAPI.listDetailsForSpecifiedProfile(token, "enclosure", 1);
System.out.println(profileObj.getName() + "\n");
```
- c. Log out from the CMS:

```
vcemAPI.logout(token);
}
}
```

listServerProfiles

Summary

This SOAP API returns a list of all configured server profiles.

Values in certain fields are not returned by this SOAP API in real time. Because no volatile information is stored in the database, you must request the information in real time through the VCM. For the bay, the transient information includes all data from the server.

API

```
ServerProfile [ ] listServerProfiles (String token)
```

Parameters

Login token

Return value

List of existing server profiles

Exceptions

2401, 2801, 2804, 2805, 2806

Minimum privileges

Read only

Sample code

From the external application computer, perform the following steps:

1. Access `https://mycms.corp.com:50001/GetCertificate`.
2. Save the certificate as `sim.cer` in the current working directory.
3. Type the following text in the command line, followed by the password:

```
keytool -import -keystore sim.ks -file sim.cer
password is "changeit"
```

4. Set a trusted certificate by entering the following SOAP API code in an external application (such as testAPI.java):


```
import java.net.URL;
public class ListServerProfilesAPISample {
    public static void main(String[] args) throws Exception {
        System.setProperty("javax.net.ssl.trustStore", "sim.ks");
        System.setProperty("javax.net.ssl.trustStorePassword", "changeit");
```
5. Create a service.


```
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator service = new
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator();
```
6. Create a stub using the service.


```
String url =
"https://mycms.corp.com:50001/mxsoap/services/mvcd1_0";
com.hp.mvcd.api.v1_0soap.SoapApi vcemAPI = service.getMvcd1_0(new
URL(url));
```
7. Execute the SOAP APIs.
 - a. Log in to the CMS:


```
String token = vcemAPI.login("username", "passwd");
```
 - b. Return a list of server profiles:


```
com.hp.mvcd.api.v1_0soap.ServerProfile[] profileList =
vcemAPI.listServerProfiles(token);
for (int i = 0; i < profileList.length; i++)
{System.out.println(profileList[i].getName() + "\n");
}
```
 - c. Log out from the CMS:


```
vcemAPI.logout(token);
}
```

getVCDomainForEnclosure

Summary

This SOAP API returns the VC Domain in which a given enclosure resides.

API

VCDomain getVCDomainForEnclosure (String token, String enclosureName)

Parameters

- Login token
- Enclosure name

Return value

VC Domain

Exceptions

2012, 2116, 2139, 2201, 2401, 2801, 2804, 2805, 2806, 5012, 5052

Minimum privileges

Read only

Sample code

From the external application computer, perform the following steps:

1. Access `https://mycms.corp.com:50001/GetCertificate`.
2. Save the certificate as `sim.cer` in the current working directory.
3. Type the following text in the command line, followed by the password:
`keytool -import -keystore sim.ks -file sim.cer`
`password is "changeit"`
4. Set a trusted certificate by entering the following SOAP API code in an external application (such as `testAPI.java`):

```
import java.net.URL;
public class GetVCDomainForEnclosureAPISample {
    public static void main(String[] args) throws Exception {
        System.setProperty("javax.net.ssl.trustStore", "sim.ks");
        System.setProperty("javax.net.ssl.trustStorePassword", "changeit");
```
5. Create a service.

```
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator service = new
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator();
```
6. Create a stub using the service.

```
String url =
"https://mycms.corp.com:50001/mxsoap/services/mvcd1_0";
com.hp.mvcd.api.v1_0soap.SoapApi vcemAPI = service.getMvcd1_0(new
URL(url));
```
7. Execute the SOAP APIs.
 - a. Log in to the CMS:

```
String token = vcemAPI.login("username", "passwd");
```
 - b. Return the VC Domain of the specified enclosure:

```
com.hp.mvcd.api.v1_0soap.VCDomain vcd =
vcemAPI.getVCDomainForEnclosure(token, "enclosure");
System.out.println(vcd.getName() + "\n");
```
 - c. Log out from the CMS:

```
vcemAPI.logout(token);
}
}
```

getVCDomainGroupForEnclosure

Summary

This SOAP API returns VC Domain Group in which a given enclosure resides.

API

```
VCDomainGroup getVCDomainGroupForEnclosure (String token, String enclosureName)
```

Parameters

- Login token
- Enclosure name

Return value

VC Domain Group

Exceptions

2401, 2801, 2804, 2805, 2806, 5012, 5053

Minimum privileges

Read only

Sample code

From the external application computer, perform the following steps:

1. Access `https://mycms.corp.com:50001/GetCertificate`.
2. Save the certificate as `sim.cer` in the current working directory.
3. Type the following text in the command line, followed by the password:

```
keytool -import -keystore sim.ks -file sim.cer  
password is "changeit"
```
4. Set a trusted certificate by entering the following SOAP API code in an external application (such as `testAPI.java`):

```
import java.net.URL;  
  
public class GetVCDomainForEnclosureAPISample {  
    public static void main(String[] args) throws Exception {  
        System.setProperty("javax.net.ssl.trustStore", "sim.ks");  
        System.setProperty("javax.net.ssl.trustStorePassword", "changeit");  
    }  
}
```
5. Create a service.

```
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator service = new  
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator();
```
6. Create a stub using the service.

```
String url =  
"https://mycms.corp.com:50001/mxsoap/services/mvcd1_0";
```



```

        com.hp.mvcd.api.v1_0soap.SoapApi vcmAPI = service.getMvcd1_0(new
        URL(url));
7.  Execute the SOAP APIs.
    a.  Log in to the CMS:
        String token = vcmAPI.login("username","passwd");
    b.  Return the VC Domain of the specified enclosure:
        com.hp.mvcd.api.v1_0soap.VCDomain vcd =
        vcmAPI.getVCDomainForEnclosure(token, "enclosure");
        System.out.println(vcd.getName() + "\n");
    c.  Log out from the CMS:
        vcmAPI.logout(token);
    }
}

```

isEnclosureLicensed

Summary

This SOAP API returns whether the enclosure is licensed.

API

```
public Boolean isEnclosureLicensed (String token, String enclosureName)
```

Parameters

- Login token
- Enclosure name

Return value

True or False

Exceptions

2116, 2138, 2139, 2201, 2801, 2804, 2805, 2806, 5012

Minimum privileges

Read only

Sample code

From the external application computer, perform the following steps:

1. Access `https://mycms.corp.com:50001/GetCertificate`.
2. Save the certificate as `sim.cer` in the current working directory.
3. Type the following text in the command line, followed by the password:


```
keytool -import -keystore sim.ks -file sim.cer
password is "changeit"
```

4. Set a trusted certificate by entering the following SOAP API code in an external application (such as testAPI.java):

```
import java.net.URL;
public class GetVCDomainForEnclosureAPISample {
    public static void main(String[] args) throws Exception {
        System.setProperty("javax.net.ssl.trustStore", "sim.ks");
        System.setProperty("javax.net.ssl.trustStorePassword", "changeit");
```

5. Create a service.

```
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator service = new
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator();
```

6. Create a stub using the service.

```
String url =
"https://mycms.corp.com:50001/mxsoap/services/mvcd1_0";
com.hp.mvcd.api.v1_0soap.SoapApi vcemAPI = service.getMvcd1_0(new
URL(url));
```

7. Execute the SOAP APIs.

- a. Log in to the CMS:

```
String token = vcemAPI.login("username", "passwd");
```

- b. Return the VC Domain of the specified enclosure:

```
com.hp.mvcd.api.v1_0soap.VCDomain vcd =
vcemAPI.getVCDomainForEnclosure(token, "enclosure");
System.out.println(vcd.getName() + "\n");
```

- c. Log out from the CMS:

```
vcemAPI.logout(token);
```

```
}
```

```
}
```

listStatusForMvcdJob

Summary

This SOAP API lists the status of a given VCEM job.

API

Job listStatusForMvcdJob (String token, Long jobID)

Parameters

- Login token
- JobID

Return value

Multiple job objects

Exceptions

2401, 2801, 2804, 2805, 2806, 5014

Minimum privileges

Read only

Objects and attributes

Job object:

- String: `jobName`
- String: `jobDescription`
- Long: `jobID`
- String: `state`
 - UNKNOWN
 - PENDING
 - RUNNING
 - COMPLETED
 - FAILED
 - CANCELLED
- String: `type`
 - MDM
 - FAILOVER
- Int: `priority`
- String: `userName`
- Long: `createTime`
- Long: `startTime`
- Long: `endTime`
- `jobProgress[]`: array of `JobProgress` that stores all the job progress information of a particular job.
 - Long: `currentTime`
 - Int: `percentComplete`
 - String: `progressDescription`

Sample code

From the external application computer, perform the following steps:

1. Access `https://mycms.corp.com:50001/GetCertificate`.
2. Save the certificate as `sim.cer` in the current working directory.
3. Type the following text in the command line, followed by the password:

```
keytool -import -keystore sim.ks -file sim.cer  
password is "changeit"
```

4. Set a trusted certificate by entering the following SOAP API code in an external application (such as testAPI.java):


```
import java.net.URL;
public class ListStatusForMvcdJobAPISample {
    public static void main(String[] args) throws Exception {
        System.setProperty("javax.net.ssl.trustStore", "sim.ks");
        System.setProperty("javax.net.ssl.trustStorePassword", "changeit");
```
5. Create a service.


```
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator service = new
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator();
```
6. Create a stub using the service.


```
String url =
"https://mycms.corp.com:50001/mxsoap/services/mvcd1_0";
com.hp.mvcd.api.v1_0soap.SoapApi vcemAPI = service.getMvcd1_0(new
URL(url));
```
7. Execute the SOAP APIs.
 - a. Log in to the CMS:


```
String token = vcemAPI.login("username","passwd");
```
 - b. Return the job object:


```
com.hp.mvcd.api.v1_0soap.Job jobObj =
vcemAPI.listStatusForMvcdJob(token, (long) 1);
```
 - c. Print the specified job state:


```
System.out.println(jobObj.getState());
```
 - d. Log out from the CMS:


```
vcemAPI.logout(token);
}
}
```

createProfile

Summary

This SOAP API creates a server profile.

If no bay information exists or if the entered bay does not support the server profile, then the SOAP API creates a new server profile, but the profile is not assigned.

If a given bay does not have a server blade, then the SOAP API creates a new server profile and assigns the profile to the bay.

API

```
Long createProfile (String token, ServerProfile profileObject)
```

Parameters

- Login token
- Serverprofile object

Return value

Long JobID

Exceptions

1508, 2401, 2551, 2552, 2556, 2559, 2601, 2602, 2607, 2611, 2615, 2801, 2802, 2804, 2805, 2806, 5017, 5019, 5020, 5022, 5024, 5026, 5027, 5028, 5029, 5030, 5031, 5032, 5033, 5034, 5035, 5036, 5037, 5048, 5049

Minimum privileges

Full access

Usage notes

- To create a profile, you must enter necessary server profile attributes.
- Each VCEM managed profile must have a unique profile name.
- If there is no existing network connection or Fibre Channel (FC) connection, set both to null within the profile object.
- Boot priority requires both a WWPN and LUN (unless Boot priority is set to disabled or use BIOS).
- You must assign a VC Domain Group where the profile will be placed.

Objects and attributes:

To create a profile, values for the following attributes must be set as passing in the parameter.

- String: name
- Long: VCDGId
- Array of NetworkConnection
 - Integer: portNumber
 - Network: network
 - String: name
 - String: pxeBootSetting
 - DISABLED
 - ENABLED
 - USE-BIOS
- Array of FCConnection
 - Integer: portNumber
 - String: bootportWWN
 - Integer: bootlun
 - String: portSpeed
 - DISABLED
 - AUTO
 - 2Gb
 - 4Gb
 - String: fcBootPriority
 - DISABLED

- PRIMARY
- SECONDARY
- USE_BIOS

NOTE: If you use PRIMARY and SECONDARY in the fcBootPriority string, then bootportWWN and bootlun must have valid values.

- o FCFabric: fcFabric
 - String: name
- o Bay:
 - Long: bayid

NOTE: To assign the new profile to a bay, you must enter bay information. Otherwise, profileObject.bay must be null.

Sample code

From the external application computer, perform the following steps:

1. Access `https://mycms.corp.com:50001/GetCertificate`.
2. Save the certificate as `sim.cer` in the current working directory.
3. Type the following text in the command line, followed by the password:


```
keytool -import -keystore sim.ks -file sim.cer
password is "changeit"
```
4. Set a trusted certificate by entering the following SOAP API code in an external application (such as `testAPI.java`):


```
import java.net.URL;
public class CreateProfileAPISample {
    public static void main(String[] args) throws Exception {
        System.setProperty("javax.net.ssl.trustStore", "sim.ks");
        System.setProperty("javax.net.ssl.trustStorePassword", "changeit");
```
5. Create a service.


```
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator service = new
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator();
```
6. Create a stub using the service.


```
String url =
"https://mycms.corp.com:50001/mxsoap/services/mvcd1_0";
com.hp.mvcd.api.v1_0soap.SoapApi vcmAPI = service.getMvcd1_0(new
URL(url));
```
7. Execute the SOAP APIs.
 - a. Log in to the CMS:


```
String token = vcmAPI.login("username","passwd");
```
 - b. Create a server profile:


```
com.hp.mvcd.api.v1_0soap.ServerProfile profile = new
com.hp.mvcd.api.v1_0soap.ServerProfile();
profile.setName("myProfile");
profile.setVCDGId((long) 1);
```
 - c. To assign a new profile, set the bay ID:

```

    com.hp.mvcd.api.v1_0soap.Bay bay = new
    com.hp.mvcd.api.v1_0soap.Bay();
    bay.setBayId((long) 1);
    profile.setBay(bay);
d. Set one network connection for the profile:
    com.hp.mvcd.api.v1_0soap.NetworkConnection networkConnection = new
    com.hp.mvcd.api.v1_0soap.NetworkConnection();
    networkConnection.setPortNumber(1);
    networkConnection.setPxeSetting("USE-BIOS");
e. Set the network used by the network connection:
    com.hp.mvcd.api.v1_0soap.Network network = new
    com.hp.mvcd.api.v1_0soap.Network();
    network.setName("network1");
f. Assign the network to the network connection, and the network connection to the profile:
    networkConnection.setNetwork(network);
    com.hp.mvcd.api.v1_0soap.NetworkConnection[] networkConnectionList
    = { networkConnection };
    profile.setNetworkConnection(networkConnectionList);
g. Set one FC connection for the profile:
    com.hp.mvcd.api.v1_0soap.FCConnection fcConnection = new
    com.hp.mvcd.api.v1_0soap.FCConnection();
    fcConnection.setPortNumber(1);
    fcConnection.setBootportWWN("50:00:1F:E1:50:09:75:38");
    fcConnection.setBootlun(1);
    fcConnection.setPortSpeed("AUTO");
    fcConnection.setFcBootPriority("PRIMARY");
h. Set the FC fabric for the FC connection:
    com.hp.mvcd.api.v1_0soap.FCFabric fcFabric = new
    com.hp.mvcd.api.v1_0soap.FCFabric();
    fcFabric.setName("san1");
i. Assign the FC fabric to the FC connection, and the FC connection to the profile:
    fcConnection.setFcFabric(fcFabric);
    com.hp.mvcd.api.v1_0soap.FCConnection[] fcConnectionList = {
    fcConnection };
    profile.setFcConnection(fcConnectionList);
j. Create the server profile above and return job ID:
    long jobID = vcemAPI.createProfile(token, profile);
k. Access the Create Profile job object:
    com.hp.mvcd.api.v1_0soap.Job jobObj =
    vcemAPI.listStatusForMvcdJob(token, jobID);
l. Print the Create Profile job state:
    System.out.println(jobObj.getState());
m. Log out from the CMS:
    vcemAPI.logout(token);
}
}

```

assignProfile

Summary

This SOAP API assigns an existing profile to a given enclosure and bay.

After using the assignProfile SOAP API, a JobID is returned. You can use the JobID with the ListStatusForMvcdJob() SOAP API to request the job status from the CMS.

API

Long assignProfile (String token, Long profileID, String enclosureName, int bayNumber)

Parameters

- Login token
- Server profile ID
- Enclosure name
- Bay number

Return value

Long JobID

Exceptions

2012, 2116, 2138, 2139, 2142, 2401, 2616, 2631, 2801, 2802, 2804, 2805, 2806, 5012, 5013, 5016, 5046

Minimum privileges

Full access

Usage notes

- If the bay has a server, you must power off the server.
- The requested bay must be associated with an enclosure.
- The requested bay must not have an assigned profile.
- The profile cannot already be associated with a bay.
- The bay must belong to the same VC Domain Group as the profile.

Sample code

From the external application computer, perform the following steps:

1. Access `https://mycms.corp.com:50001/GetCertificate`.
2. Save the certificate as `sim.cer` in the current working directory.
3. Type the following text in the command line, followed by the password:

```
keytool -import -keystore sim.ks -file sim.cer  
password is "changeit"
```


4. Set a trusted certificate by entering the following SOAP API code in an external application (such as testAPI.java):

```
import java.net.URL;
import com.hp.mvcd.api.v1_0soap.Job;
import com.hp.mvcd.api.v1_0soap.ServerProfile;

public class AssignProfileAPISample {
    public static void main(String[] args) throws Exception {
        System.setProperty("javax.net.ssl.trustStore", "sim.ks");
        System.setProperty("javax.net.ssl.trustStorePassword", "changeit");
```

5. Create a service.

```
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator service = new
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator();
```

6. Create a stub using the service.

```
String url =
"https://mycms.corp.com:50001/mxsoap/services/mvcd1_0";
com.hp.mvcd.api.v1_0soap.SoapApi vcmAPI = service.getMvcd1_0(new
URL(url));
```

7. Execute the SOAP APIs.

- a. Log in to the CMS:

```
String token = vcmAPI.login("username","passwd");
```

- b. Return a list of server profiles:

```
ServerProfile[] profileList = vcmAPI.listServerProfiles(token);
long profileID = 0;
```

- c. Search for the "myProfile" profile ID:

```
for (int i = 0; i < profileList.length; i++) {
    if (profileList[i].getName().equals("myProfile")) {
        profileID = profileList[i].getId();
    }
}
```

- d. Assign the server profile to bay 1 on an enclosure named "enclosure":

```
long jobID = vcmAPI.assignProfile(token, profileID, "enclosure",
1);
```

- e. Return the Assign Profile job object:

```
Job jobObj = vcmAPI.listStatusForMvcdJob(token, jobID);
```

- f. Print Assign Profile job state:

```
System.out.println(jobObj.getState());
```

- g. Log out from the CMS:

```
vcmAPI.logout(token);
```

```
    }
}
```

migrateProfile

Summary

This SOAP API migrates an existing profile to a given enclosure and bay.

After using the migrateProfile SOAP API, a JobID is returned. You can use the JobID with the ListStatusForMvcdJob() SOAP API to request the job status from the CMS.

API

Long migrateProfile (String token, Long profileID, String enclosureName, int bayNumber)

Parameters

- Login token
- Server profile ID
- Enclosure name
- Bay number

Return value

Long JobID of the migration job

Exceptions

1508, 2012, 2116, 2138, 2139, 2142, 2401, 2624, 2631, 2801, 2802, 2804, 2805, 2806, 5012, 5013, 5016, 5046

Minimum privileges

Full access

Usage notes

- All domains must belong to the same VC Domain Group.
- If a bay involved in the migration has a server, you must power off the server.
- The destination bay must not have a server profile assigned.
- VCEM cannot prevent a migration based on source or destination server models.
- VCEM can prevent a migration to impossible choices, such as a storage blade.

Sample code

From the external application computer, perform the following steps:

1. Access `https://mycms.corp.com:50001/GetCertificate`.
2. Save the certificate as `sim.cer` in the current working directory.
3. Type the following text in the command line, followed by the password:

```
keytool -import -keystore sim.ks -file sim.cer  
password is "changeit"
```

4. Set a trusted certificate by entering the following SOAP API code in an external application (such as testAPI.java):


```
import java.net.URL;
public class MigrateProfileAPISample {
    public static void main(String[] args) throws Exception {
        System.setProperty("javax.net.ssl.trustStore", "sim.ks");
        System.setProperty("javax.net.ssl.trustStorePassword", "changeit");
```
5. Create a service.


```
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator service = new
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator();
```
6. Create a stub using the service.


```
String url =
"https://mycms.corp.com:50001/mxsoap/services/mvcd1_0";
com.hp.mvcd.api.v1_0soap.SoapApi vcemAPI = service.getMvcd1_0(new
URL(url));
```
7. Execute the SOAP APIs.
 - a. Log in to the CMS:


```
String token = vcemAPI.login("username", "passwd");
```
 - b. Return a list of server profiles:


```
com.hp.mvcd.api.v1_0soap.ServerProfile[] profileList =
vcemAPI.listServerProfiles(token);
long profileID = 0;
```
 - c. Search for the specified profile ID:


```
for (int i = 0; i < profileList.length; i++) {
    if (profileList[i].getName().equals("myProfile")) {
        profileID = profileList[i].getId();
    }
}
```
 - d. Migrate the server profile to bay 7 on an enclosure named "enclosure":


```
long jobID = vcemAPI.migrateProfile(token, profileID, "enclosure",
7);
System.out.println(jobID);
```
 - e. Return the Migrate Profile job object:


```
com.hp.mvcd.api.v1_0soap.Job jobObj =
vcemAPI.listStatusForMvcdJob(token, jobID);
```
 - f. Print Migrate Profile job state:


```
System.out.println(jobObj.getState());
```
 - g. Log out from the CMS:


```
vcemAPI.logout(token);
}
}
```

checkPowerStatusOfBay

Summary

This SOAP API returns the power status of a given bay. If no blade is present in the bay, the SOAP API returns a status of UNKNOWN.

API

```
String checkPowerStatusOfBay(String token, String enclosureName, int bayNumber)
```

Parameters

- Login token
- Enclosure name
- Bay number

Return value

On, Off, Unknown

Exceptions

1126, 2012, 2116, 2138, 2139, 2142, 2401, 2801, 2804, 2805, 2806, 5012, 5013, 5046

Minimum privileges

Read only

Sample code

From the external application computer, perform the following steps:

1. Access `https://mycms.corp.com:50001/GetCertificate`.
2. Save the certificate as `sim.cer` in the current working directory.
3. Type the following text in the command line, followed by the password:

```
keytool -import -keystore sim.ks -file sim.cer  
password is "changeit"
```
4. Set a trusted certificate by entering the following SOAP API code in an external application (such as `testAPI.java`):

```
import java.net.URL;  
public class CheckPowerStatusOfBayAPISample {  
    public static void main(String[] args) throws Exception {  
        System.setProperty("javax.net.ssl.trustStore", "sim.ks");  
        System.setProperty("javax.net.ssl.trustStorePassword", "changeit");  
    }  
}
```
5. Create a service.

```
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator service = new  
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator();
```
6. Create a stub using the service.

```
String url =
    "https://mycms.corp.com:50001/mxsoap/services/mvcd1_0";
com.hp.mvcd.api.v1_0soap.SoapApi vcmAPI = service.getMvcd1_0(new
URL(url));
```

7. Execute the SOAP APIs.
 - a. Log in to the CMS:

```
String token = vcmAPI.login("username", "passwd");
```
 - b. Verify the power status of the specified bay (bay 1) on an enclosure named "enclosure":

```
String powerStatus = vcmAPI.checkPowerStatusOfBay(token,
"enclosure", 1);
System.out.println(powerStatus);
```
 - c. Log out from the CMS:

```
vcmAPI.logout(token);
}
}
```

powerBayOn

Summary

This SOAP API sends a power-on request to the Virtual Connect Manager.

API

```
void powerBayOn (String token, String enclosureName, int bayNumber)
```

Parameters

- Login token
- Enclosure name
- Bay number

Return value

Void

Exceptions

1119, 1126, 2012, 2116, 2138, 2139, 2142, 2146, 2401, 2801, 2802, 2804, 2805, 2806, 5012, 5013, 5046, 5051

Minimum privileges

Full access

Usage notes

The bay must have a server.

Sample code

From the external application computer, perform the following steps:

1. Access `https://mycms.corp.com:50001/GetCertificate`.
2. Save the certificate as `sim.cer` in the current working directory.
3. Type the following text in the command line, followed by the password:
`keytool -import -keystore sim.ks -file sim.cer`
`password is "changeit"`
4. Set a trusted certificate by entering the following SOAP API code in an external application (such as `testAPI.java`):

```
import java.net.URL;
public class PowerBayOnAPISample {
    public static void main(String[] args) throws Exception {
        System.setProperty("javax.net.ssl.trustStore", "sim.ks");
        System.setProperty("javax.net.ssl.trustStorePassword", "changeit");
    }
}
```
5. Create a service.

```
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator service = new
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator();
```
6. Create a stub using the service.

```
String url =
"https://mycms.corp.com:50001/mxsoap/services/mvcd1_0";
com.hp.mvcd.api.v1_0soap.SoapApi vcmAPI = service.getMvcd1_0(new
URL(url));
```
7. Execute the SOAP APIs.
 - a. Log in to the CMS:

```
String token = vcmAPI.login("username","passwd");
```
 - b. Power up the specified bay (bay 1) on an enclosure named "enclosure":

```
vcmAPI.powerBayOn(token, "enclosure", 1);
```
 - c. Log out from the CMS:

```
vcmAPI.logout(token);
}
}
```

powerBayOff

Summary

This SOAP API sends a power-off request to the VCM.

If the parameter `graceful` is entered as `true`, the bay powers off gracefully. If this parameter is entered as `false`, a hard shutdown occurs. To verify the status of the bay, use `checkPowerStatusOfBay`.

API

```
void powerBayOff (String token, String enclosureName, int bayNumber, boolean
graceful)
```

Parameters

- Login token
- Enclosure name
- Bay number
- graceful

Return value

Void

Exceptions

1119, 1126, 2012, 2116, 2138, 2139, 2142, 2146, 2401, 2801, 2802, 2804, 2805, 2806, 5012, 5013, 5046, 5050

Minimum privileges

Full access

Usage notes

The bay must have a server.

Sample code

From the external application computer, perform the following steps:

1. Access `https://mycms.corp.com:50001/GetCertificate`.
2. Save the certificate as `sim.cer` in the current working directory.
3. Type the following text in the command line, followed by the password:

```
keytool -import -keystore sim.ks -file sim.cer  
password is "changeit"
```
4. Set a trusted certificate by entering the following SOAP API code in an external application (such as `testAPI.java`):

```
import java.net.URL;  
public class PowerBayOffAPISample {  
    public static void main(String[] args) throws Exception {  
        System.setProperty("javax.net.ssl.trustStore", "sim.ks");  
        System.setProperty("javax.net.ssl.trustStorePassword", "changeit");  
    }  
}
```
5. Create a service.

```
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator service = new  
com.hp.mvcd.api.v1_0soap.SoapApiServiceLocator();
```
6. Create a stub using the service.

```
String url =  
"https://mycms.corp.com:50001/mxsoap/services/mvcd1_0";  
com.hp.mvcd.api.v1_0soap.SoapApi vcemAPI = service.getMvcd1_0(new  
URL(url));
```
7. Execute the SOAP APIs.
 - a. Log in to the CMS:

```
String token = vcemAPI.login("username","passwd");  
b. Power down the specified bay (bay 1) on an enclosure named "enclosure":  
    vcemAPI.powerBayOff(token, "enclosure", 1, false);  
c. Check the power status of the specified bay (bay 1) on an enclosure named "enclosure":  
    String powerStatus = vcemAPI.checkPowerStatusOfBay(token,  
    "enclosure", 1);  
    System.out.println(powerStatus);  
d. Log out from the CMS:  
    vcemAPI.logout(token);  
    }  
}
```

Implementation

This chapter provides additional information on SOAP APIs and lists fault exceptions generated when errors occur.

WSDL

The Web Services Description Language (WSDL) file is an Extensible Markup Language (XML) file that describes the available SOAP APIs. The WSDL file is an important file that you can consult to verify available SOAP APIs.

To access the WSDL, access the following URL:

```
https://<CMS_hostname>:50001/mxsoap/services/mvcd1_0?wsdl
```

To access the endpoint of the WSDL, enter:

```
https://<CMS_hostname>:50001/mxsoap/services/mvcd1_0
```

NOTE: If your SOAP port number is no longer 50001, then update these URLs accordingly. For more information on ports used, see the “Ports” section in this guide.

Fault exceptions

If you encounter an error when using a SOAP API, then a fault exception is generated. The following table lists the fault exceptions that you can encounter when using SOAP APIs.

Table 1 SOAP API fault exceptions

Exception code	Fault exception
1119	Failed to perform a power operation. Virtual Connect Enterprise Manager cannot retrieve the information of the server associated with the Server Bay. Verify that the server is properly inserted in the Server Bay.
1126	Virtual Connect Enterprise Manager cannot get the Virtual Connect username and password from mxpi.
1508	Virtual Connect Manager is not responding. It could possibly be 1. The IP address is no longer active. 2. Network issues associated with the Virtual Connect Manager. The corrective actions are: 1. In a redundant environment, check if failover between active and standby Virtual Connect module occurs. If so, wait until failover complete and try again. 2. Verify that you can ping from the command line of the Virtual Connect Enterprise Manager server and log into the Virtual Connect Manager through the browser.
2012	The operation cannot be performed because the VC Domain [{0}] does not have a relationship with an enclosure in HP Systems Insight Manager (HP SIM). Verify the VC Domain was correctly discovered by HP SIM, or run HP SIM discovery again and retry the operation.
2116	VCEM could not verify the license of the enclosure [{0}] in the specified VC Domain because it was not possible to retrieve license information from HP Systems Insight Manager (HP SIM). Verify HP SIM is running and then perform the operation again.

Table 1 SOAP API fault exceptions

Exception code	Fault exception
2138	VCEM could not verify the license of the specified VC Domain because the enclosure [{0}] does not have a name or UUID and they are required.
2139	VCEM could not verify the license for enclosure [{0}] in the specified VC Domain because it failed to communicate with HP Systems Insight Manager (HP SIM). Verify HP SIM is correctly functioning and perform the operation again.
2142	VCEM could not verify the license because the specified enclosures list is empty.
2145	VCEM could not retrieve the product version. Verify VCEM was correctly installed and is running over a supported platform.
2146	VCEM cannot perform the requested power operation because the current power state of the server is [{0}].
2201	There was an error getting the VC Domain by the primary enclosure name [{0}] from HP Systems Insight Manager. The enclosure name is not valid.
2401	Error on database operation. Check that VCEM was correctly installed and that the database service is running.
2551	Cannot perform server profile operation because the server profile name is invalid. Names can only have alphanumeric characters, hyphens (-) or underbars (_) and are limited to 64 characters. Enter a well formed name and then perform the operation again.
2552	Cannot create server profile because a server profile with the same name already exists in VCEM. Enter a unique name for the server profile and then perform the operation again.
2556	Cannot create network connection because the specified network does not exist. Enter a valid network name.
2559	Cannot create a network connection for the entered port. Ethernet network connections require a valid network. Verify all port connections are correctly pointing to a valid network and then perform the operation again.
2601	Cannot create FC SAN connection because FC port is outside the allowed range.
2602	Cannot create FC SAN connection because the specified FC SAN fabric does not exist.
2607	Cannot set SAN boot settings because the entered LUN is not valid. Valid values are from 0 to 255.
2611	Cannot create server profile because the specified VC Domain Group does not exist. A server profile must be assigned to an existent VC Domain Group.
2615	Cannot perform server profile assignment because the specified target Server Bay is not valid. Verify the specified target Server Bay belongs to a managed VC Domain.
2616	Cannot perform server profile assignment because the server profile is already assigned to another Server Bay.
2624	Cannot move the server profile because the specified server profile is not currently assigned to a Server Bay.
2631	Specified server profile ID does not correspond to a managed server profile in VCEM.

Table 1 SOAP API fault exceptions

Exception code	Fault exception
2801	The user session is expired. Login again to obtain a new session and perform the operation again.
2802	This action cannot be performed because the current user has read-only privileges.
2803	Error logging in. Incorrect username [{0}] or password.
2804	Access denied. The entered logon token is invalid. Obtain a new token and perform the operation again.
2805	Error logging in: {0}
2806	This user does not have authorization to use or view HP Virtual Connect Enterprise Manager.
2807	The entered logon token is invalid. Obtain a new token and perform the operation again.
2808	Error logging out: {0}
5010	User name cannot be empty.
5011	Password cannot be empty.
5012	Enclosure Name cannot be empty.
5013	Specified Server Bay number is out of range, it must be between 1 ~ 16.
5014	No Job is found. Verify that the specified Job ID is valid and greater than 0.
5016	Specified server profile ID must be bigger than 0.
5017	Specified FCBootPriority, FCPortSpeed or PXEBootSetting type is invalid to create a server profile.
5019	Specified LUN number is invalid. It has to be a number between 0 and 255.
5020	Specified WWN is invalid. It has to be in the format of mm-mm-mm-mm-mm-mm-mm-mm.
5021	Specified VC Domain ID must be greater than 0
5022	Specified VC Domain Group ID must be greater than 0
5024	Server profile cannot have more than 16 Network connections.
5026	Specified profile name cannot be empty
5027	Invalid network connection port number. Port number must be unique for each network connection.
5028	Duplicate network connection port number. Port number must be greater than 0.
5029	Invalid network connection port number. Port number cannot be empty.
5030	Invalid network connection PXE setting. PXE setting cannot be empty.
5031	Invalid network connection PXE setting. PXE setting cannot be set ENABLE for more than one network connection.
5032	Invalid FC connection port number. Port number must be greater than 0.
5033	Invalid FC connection port number. Port number must be unique for each FC connection.

Table 1 SOAP API fault exceptions

Exception code	Fault exception
5034	Invalid FC connection port number. Port number cannot be empty
5036	Invalid FC connection WWN. WWN cannot be empty when Boot Priority is Primary or Secondary
5037	Server profile object cannot be null.
5045	Couldn't locate the specified profile. Verify the enclosure name and bayNumber are valid.
5046	Couldn't locate the specified Server Bay. Verify the enclosure name and bay number are valid.
5048	Server profile was not assigned to the specified Server Bay, because the Server Bay already has a server profile assigned.
5049	No valid Server Bay is found. Verify that the specified Server Bay ID is correct.
5050	Specified Server Bay is already powered off.
5051	Specified Server Bay is already powered on.
5052	VC Domain associated with the enclosure name was not found. Verify that the enclosure name is valid.
5053	VC Domain Group associated with the enclosure name was not found. Verify that the enclosure name is valid.
5054	Server Profile associated with the VC Domain ID was not found. Verify that the VC Domain ID is valid.
5055	Server Profile associated with the VC Domain Group ID was not found. Verify that the VC Domain Group ID is valid.

HP services and technical support

HP offers a number of software support services, many of which are provided to our customers at no additional charge.

- Software Technical Support and Update Service—VCEM includes one year of 24 x 7 HP Software Technical Support and Update Service. This service provides access to HP technical resources for assistance in resolving software implementation or operations problems. The service also provides access to software updates and reference manuals either in electronic form or on physical media as they are made available from HP. (Customers who purchase an electronic license to use are eligible for electronic updates only.) With this service, customers will benefit from expedited problem resolution as well as proactive notification and delivery of software updates. For more information about this service, see <http://www.hp.com/services/insight>.

Registration for Software Technical Support and Update Service:

There are two methods for registering:

- If you received a license entitlement certificate, automated registration for this service will take place upon online redemption of the license certificate/key.
- If the license information you received for your product instructs you to register for Software Technical Support and Update Service, follow the instructions so that you will be eligible for telephone support and product updates.

How to Use Your Software Technical Support and Update Service:

After registration, you will receive a service contract in the mail containing the Customer Service phone number and your Service Agreement Identifier (SAID). You will need your SAID when calling for technical support. Using your SAID, you can also go to the Software Update Manager (SUM) web page to view your contract online and elect electronic delivery for product updates.

- Warranty—HP will replace defective delivery media for a period of 90 days from the date of purchase. This warranty applies to VCEM and all Insight Control Management, HP Systems Insight Manager, and ProLiant Essentials products.
- Join the discussion—The HP Support Forum is a community-based, user-supported tool for HP customers to participate in discussions amongst the customer community about HP products. See the “Management Software and System Tools” area.
- Software and Drivers download pages—These pages provide the latest software and drivers for your HP products.
- Management Security (<http://www.hp.com/servers/manage/security>)—HP is proactive in its approach to the quality and security of all its management software. Be sure to check this website often for the latest down loadable security updates.

HP Worldwide Customer Service contact numbers are available at <http://www.hp.com/country/us/en/wwcontact.html>. For U.S. customers, say “Insight Manager” when prompted for the product name.

Where to go for additional help

In addition to this guide, the following resources are available:

- <http://www.hp.com/go/vcem>
- *HP Virtual Connect Enterprise Manager User Guide*
- *HP Virtual Connect Enterprise Manager Release Notes*
- *HP Virtual Connect Enterprise Manager Support Matrix*

For more information about Virtual Connect Manager, see the following resources:

- *HP BladeSystem c-Class Solution Overview*
- *HP Virtual Connect for c-Class BladeSystem User Guide*
- *HP Virtual Connect Manager Release Notes*

HP contact information

For the name of the nearest HP authorized reseller:

- In the United States, see the HP U.S. service locator at http://www.hp.com/service_locator.
- In other locations, see Contact HP Worldwide at <http://welcome.hp.com/country/us/en/wwcontact.html>.

For HP technical support:

- In the United States, for contact options see Contact HP United States at http://welcome.hp.com/country/us/en/contact_us.html. To contact HP by phone, call 1-800-HP-INVENT (1-800-474-6836). This service is available 24 hours a day, 7 days a week. For continuous quality improvement, calls may be recorded or monitored. When prompted, say “Insight Manager.”
- In other locations, see Contact HP Worldwide at <http://welcome.hp.com/country/us/en/wwcontact.html>.