Universal CMDB

Software Version: 10.30

JMX Reference Guide

Document Release Date: October 2016
Software Release Date: October 2016

Hewlett Packard Enterprise

Legal Notices

Warranty

Restricted Rights Legend

Copyright Notice

Trademark Notices

Documentation Updates

To check for recent updates or to verify that you are using the most recent edition of a document, go to: https://softwaresupport.hpe.com/.

This site requires that you register for an HP Passport and to sign in. To register for an HP Passport ID, click **Register** on the HPE Software Support site or click **Create an Account** on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HPE sales representative for details.

Support

Visit the HPE Software Support site at: https://softwaresupport.hpe.com/.

This website provides contact information and details about the products, services, and support that HPE Software offers.

HPE Software online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support website to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HPE support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and to sign in. Many also require a support contract. To register for an HP Passport ID, click **Register** on the HPE Support site or click **Create an Account** on the HP Passport login page.

To find more information about access levels, go to: https://softwaresupport.hp.com/web/softwaresupport/access-levels.

HPSW Integrations Catalog accesses the new HPSW Integrations and Solutions Catalog website. This site enables you to explore HPE Product Solutions to meet your business needs, includes a full list of Integrations between HPE Products, as well as a listing of ITIL Processes. The URL for this website is https://softwaresupport.hpe.com/km/KM01702731.

Contents

Chapter 1: Introduction

This chapter includes:

Introduction

This guide provides a reference for JMX methods included in the UCMDB documentation. Many UCMDB actions can be performed from the JMX console. You can search the JMX Quick Search page for JMX methods as described below.

> **Note:** The methods in this guide were collected from existing documentation.

UCMDB JMX Console

1. On the UCMDB server, launch the Web browser and enter the following address: **https://localhost:8443/jmx-console**.

   > **Note:** Starting from version 10.30, access to the JMX console is restricted to localhost only. If you need to access the JMX console remotely, see "How to Enable Remote Access to the JMX Console" in the *HPE Universal CMDB Administration Guide*.

2. Enter the JMX console authentication credentials (Login name = **sysadmin**).

   The UCMDB JMX Quick Search page opens. There are three ways to access a JMX operation from the JMX home page.

   ○ **Use the JMX quick search**

     The JMX quick search feature provides the ability to:

- Search for a service. This is useful when you know that an operation is in a certain service category, but you do not know the name of the operation.

- Search for a JMX operation based on a keyword

  - Keywords can be an operation name, the description of the operation, or even the parameters used by the operation.

  - When typing, a suggestion list is displayed, providing links to quickly access suggested methods.

- Search and access a UCMDB server log from the JMX console

  - Typing the word **log:** in the search text displays a suggestion list with all the logs that contain the search word.

  - Clicking one of the suggested logs redirects to a new page displaying the full content of the log.

- **Use the UCMDB JMX link**

  Do the following:

  i. Click the UCMDB JMX link to open the console.

  ii. Locate the required service and click the link to open the operations page.

  iii. Select the required operation.

- **Use the JMX Operations Index link**

  Do the following:

  i. Click the UCMDB JMX Operations Index link to open the console operation index.

  ii. Go directly to the required method and select it.

> **Note:** It is recommended to change the JMX password. For details, see "Change the JMX Console Password" on the next page.

Data Flow Probe JMX Console

1. On the probe machine, launch the Web browser and enter the following address: **http://<Data Flow Probe machine name or IP address>:1977**. If you are running the Data Flow Probe locally, enter **http://localhost:1977**.

2. Enter the JMX console authentication credentials, which by default are:

○ Login name = **sysadmin**

○ Password = **sysadmin**

The Data Flow Probe JMX Quick Search page opens.

To search for a JMX method, enter a method name or part of a method name in the search box. The search results display all methods containing the search phrase.

3. Click the **Data Flow Probe JMX** link to open the console. Locate the required service and click the link to open the operations page. Select the required operation.

4. Click the **Data Flow Probe Operations Index** link to open the console operation index. Go directly to the required method and select it.

> **Note:** It is recommended to change the JMX password. For details, see "Change the JMX Console Password" below.

Change the JMX Console Password

1. Log in to UCMDB with an administrator account and go to **Administration > Security > Users and Groups**.

2. Select the user for the JMX Console login (by default, **sysadmin**) and click the **Reset Password** button.

3. In the Reset Password dialog box, enter the new password and confirm it. Click **OK**.

   > **Note:** The default password policy requires the password to include at least one of each of the four following types of characters:
   >
   > ○ Uppercase alphabetic characters
   >
   > ○ Lowercase alphabetic characters
   >
   > ○ Numeric characters
   >
   > ○ Symbol characters ,\:/. _?&%=+-[]()|
   >
   > It also requires the password to adhere to the minimum length, which is set by the **Password minimum length** infrastructure setting.

4. Log out of UCMDB and log in to the JMX Console using the new password.

Configuration Manager JMX Console

There is a separate JMX console for Configuration Manager.

On the Configuration Manager server, enter the following address: **http://<server name>:<application_port>/cnc/jmx-console**. The port is the port configured during the installation of Configuration Manager.

For details, see the interactive *HPE Universal CMDB Deployment Guide*.

For details on accessing the Configuration Manager JMX Console, see "Configuration Manager JMX Methods" in the *HPE Universal CMDB JMX Reference Guide*.

Java JMX Access Hardening

> **Note:** The procedure described here can also be used for the Data Flow Probe JMX.

In order to ensure that the JMX RMI port is accessible only when providing user credentials, perform the following procedure:

1. In the **wrapper.conf** file on the server, located at **C:\hp\UCMDB\UCMDBServer\bin**, set the following:

   wrapper.java.additional.16=-Dcom.sun.management.jmxremote.authenticate=true

   This setting requires the JMX to ask for authentication.

   - **For the Data Flow Probe JMX**, perform the following:

     In the files **WrapperGateway.conf** and **WrapperManager.conf**, located at **C:\hp\UCMDB\DataFlowProbe\bin**, set the following:

     wrapper.java.additional.17=-Dcom.sun.management.jmxremote.authenticate=true

2. Rename the file **jmxremote.password.template** (located at: **C:\hp\UCMDB\UCMDBServer\bin\jre\lib\management**) to **jmxremote.password**.

   > **Note:** For the Data Flow Probe JMX, this file is located at: **C:\hp\UCMDB\DataFlowProbe\bin\jre\lib\management**.

3. In **jmxremote.password**, add passwords for the roles **monitorRole** and **controlRole**.

   For example:

monitorRole QED

controlRole R&D

would assign the password **QED** to **monitorRole** and the password **R&D** to **controlRole**.

> **Note:** Ensure that only the owner has read and write permissions on **jmxremote.password** because it contains the passwords in clear text. The file owner must be the same user under which UCMDB Server is running.

4. In the file **jmxremote.access** (located at **C:\hp\UCMDB\UCMDBServer\bin\jre\lib\management**), assign access to **monitorRole** and **controlRole**.

   For example:

   monitorRole readonly

   controlRole readwrite

   would assign read-only access to **monitorRole** and read-write access to **controlRole**.

   > **Note:** For the Data Flow Probe JMX, this file is located at: **C:\hp\UCMDB\DataFlowProbe\bin\jre\lib\management**.

5. Secure files as follows:

   ○ **For Windows only:** Run the following commands from the command line to secure files:

   icacls jmxremote.password /grant Administrator:F

   icacls jmxremote.access /grant Administrator:R

   where **<username>** is the file owner visible in the properties of both files. Open properties of these files and ensure that they are correct and have only one owner.

   ○ **For Solaris and Linux operating systems:** Set the file permissions for the password file by running:

   chmod 600 jmxremote.password

6. **For Service Pack upgrades, Server migrations and Disaster Recovery:** Change ownership of the file **jmxremote.access** (located at **C:\hp\UCMDB\UCMDBServer\bin\jre\lib\management**) to the operating system user running the upgrade or migration installation.

   > **Note:**

- For the Data Flow Probe JMX, this file is located at:
  C:\hp\UCMDB\DataFlowProbe\bin\jre\lib\management.

- Before uninstalling the product, edit the file permissions for **<UMCDB installation folder>\bin\jre\lib\management\jmxremote.password** so the user you are logged in with can edit it.

UCMDB JMX Methods

Service	Method	Link to document
Authorization Services	createUser	"How to Create an Integration User" on page 97
	grantRolesToUserForAllTenants	"How to Create an Integration User" on page 97
	grantRolesToUserForTenants	"How to Create an Integration User" on page 97
	removeUser	"How to Create an Integration User" on page 97
	resetPassword	"How to Create an Integration User" on page 97 "How to Change the System User Name or Password through the JMX Console" on page 107
	setRolesForUser	"How to Create an Integration User" on page 97
	UserAuthenticate	"How to Create an Integration User" on page 97
Class Model Services	exportClassModelToXml	"How to Export the Class Model to XML" on page 64
DAL Services	getDbContext	"How to Check the Database Connection" on page 38
	modifyCompositeIndexes	"How to Modify Composite Indexes" on page 63
	rebuildIndexes	"How to Rebuild Indexes on Microsoft SQL Databases" on page 62

Service	Method	Link to document
	rebuildModelDBSchemaAndViews	"How to Rebuild the Database in Case of an Error" on page 61
	rebuildModelViews	"How to Rebuild the Database in Case of an Error" on page 61
Discovery Manager	changeEncryptionKey	"How to Generate or Update the Encryption Key for Confidential Manager" on page 125
	changeEncryptionKey	"High Availability Mode JMX Methods" on page 47
	cleanCredentialsAndRangesInformation	"How to Remove Credential and Range Information by Domain Name" on page 125
	exportCredentialsAndRangesInformation	"How to Export and Import Credential and Range Information in Encrypted Format" on page 124
	getDiscoveryStateForInventoryCI	"How to View Discovery Status of an Inventory CI in JMX" on page 77
	getAgentDeploymentLogForInventoryCI	"How to View Agent Deployment Log for an Inventory CI in JMX" on page 80
	generateEncryptionKey	"How to Generate or Update the Encryption Key for Confidential Manager" on page 125
	generateEncryptionKey	"High Availability Mode JMX Methods" on page 47
	importCredentialsAndRangesInformation	"How to Export and Import Credential and Range Information in Encrypted Format" on page 124
	importMigrationDataFromDDMI	"How to Migrate DDMI Server Configuration Data to Universal Discovery" on page 153
	recalculateAndUpdateDFMTasks	"How to Run Data Flow Ad Hoc Updates" on page 81
	recalculateAndUpdateDFMTasksForAdapter	"How to Run Data Flow Ad Hoc Updates" on page 81
	setSharedKey	"How to Set Shared Key for Encrypting or Decrypting the

Service	Method	Link to document
		InfrastructureSettings.xml File Using JMX" on page 150
High Availability Services	changeClusterAuthenticationKeystorePassword	"High Availability Mode JMX Methods" on page 47
	changeClusterEncryptionKeystorePassword	"High Availability Mode JMX Methods" on page 47
LDAP Services	configureLDAP	"How to Enable and Define the LDAP Authentication Method" on page 141
	getLDAPSettings	"How to Enable and Define the LDAP Authentication Method" on page 141
	testLDAPConnection	"How to Test LDAP Connections" on page 141
	verifyLDAPCredentials	"How to Enable and Define the LDAP Authentication Method" on page 141
Licensing Services	addLicense	"How to Manage UCMDB Licenses Using the JMX Console" on page 25
LW-SSO Configuration	addTrustedDomains	"How to Enable Login to HP Universal CMDB with LW-SSO" on page 140
	setEnabledForUI	"How to Enable Login to HP Universal CMDB with LW-SSO" on page 140
	setDomain	"How to Enable Login to HP Universal CMDB with LW-SSO" on page 140
	addTrustedIPs	"UCMDB Browser JMX Methods" on page 50
	setValidationPointHandlerEnable	"How to Enable Login to HP Universal CMDB with LW-SSO" on page 140
	updateReverseProxy	"How to Enable Login to HP Universal CMDB with LW-SSO" on page 140

Service	Method	Link to document
	setReverseProxyIPs	"How to Enable Login to HP Universal CMDB with LW-SSO" on page 140
	retrieveConfigurationFromSettings	"How to Enable Login to HP Universal CMDB with LW-SSO" on page 140
	retrieveConfiguration	"How to Enable Login to HP Universal CMDB with LW-SSO" on page 140
	setInitString	"How to Configure LW-SSO Settings" on page 119 "How to Enable Login to HP Universal CMDB with LW-SSO" on page 140
Multiple CMDB Instances Services	fetchAllDataFromAnotherCMDB	"How to Perform Initial UCMDB-UCMDB Synchronization " on page 83
	getGlobalIdGeneratorScopes	"How to Configure Global ID Generation" on page 82
	setAsGlobalIdGenerator	"How to Configure Global ID Generation" on page 82
	setAsGlobalIdGeneratorForScopes	"How to Configure Global ID Generation" on page 82
	setAsNonGlobalIdGenerator	"How to Configure Global ID Generation" on page 82
Packaging Services	deployPackages	"Package Manager JMX Methods" on page 54
	displayDeployedPackages	"Package Manager JMX Methods" on page 54
	displayResourcesDeploymentHistory	"Package Manager JMX Methods" on page 54
	exportPacakges	"Package Manager JMX Methods" on page 54
	undeployPacakges	"Package Manager JMX Methods" on page 54
Ports Managem	ComponentsConfigurations	"How to Map the UCMDB Web

Service	Method	Link to document
ent Services		Components to Ports" on page 113
		"How to Configure CAC Support on UCMDB" on page 130
		"How to Configure CAC Support for UCMDB by Reverse Proxy" on page 132
	HTTPSClientAuthSetEnable	"How to Enable or Disable HTTP/HTTPS Ports" on page 113
	HTTPSClientAuthSetPort	"How to Map the UCMDB Web Components to Ports" on page 113
	HTTPSetEnable	"How to Enable or Disable HTTP/HTTPS Ports" on page 113
	HTTPSetPort	"How to Map the UCMDB Web Components to Ports" on page 113
	HTTPSSetEnable	"How to Enable or Disable HTTP/HTTPS Ports" on page 113
	HTTPSSetPort	"How to Map the UCMDB Web Components to Ports" on page 113
	mapComponentToConnectors	"How to Map the UCMDB Web Components to Ports" on page 113
		"How to Configure CAC Support on UCMDB" on page 130
		"How to Configure CAC Support for UCMDB by Reverse Proxy" on page 132
		"How to Configure CAC (Smart Card / PKI Authentication) Support for the Embedded UCMDB Browser" on page 151
		"How to Harden the Data Flow Probe Connector in UCMDB" on page 138
		"How to Enable Mutual Certificate Authentication for SDK" on page 108
		"How to Enable Secure Login for the JMX Console" on page 147

Service	Method	Link to document
	PortsDetails	"How to Enable Mutual Certificate Authentication for SDK" on page 108 "How to Harden the Data Flow Probe Connector in UCMDB" on page 138
	serverComponentsNames	"How to Map the UCMDB Web Components to Ports" on page 113
Security Services	changeKeystorePassword	"How to Change the Server Keystore Password" on page 111
	changeMasterKeyForCluster	"How to Set Master Keys" on page 33
	CMAddUser	"How to Configure the HP Universal CMDB Server with Confidential Manager" on page 145
	CMGetConfiguration	"How to Configure Confidential Manager Communication Encryption " on page 119 "How to Configure the HP Universal CMDB Server with Confidential Manager" on page 145
	CMSetConfiguration	"How to Configure Confidential Manager Communication Encryption " on page 119 "How to Configure the HP Universal CMDB Server with Confidential Manager" on page 145
	loginWithCAC	"How to Configure CAC Support on UCMDB" on page 130 "How to Configure CAC Support for UCMDB by Reverse Proxy" on page 132
	onlyCACCerts	"How to Configure CAC Support on UCMDB" on page 130 "How to Configure CAC Support for UCMDB by Reverse Proxy" on page 132

Service	Method	Link to document
	pathToCRL	"How to Configure CAC Support on UCMDB" on page 130
	retrieveLWSSOConfiguration	"How to Retrieve the Current LW-SSO Configuration in a Distributed Environment" on page 118
	usernameField	"How to Configure CAC Support on UCMDB" on page 130
	withReverseProxy	"How to Configure CAC Support for UCMDB by Reverse Proxy" on page 132
Server Services	executeLogGrabber	"How to Download a Zip File of Log Files and Thread Dumps" on page 26
	loggersLevels	"How to Use the User Activity Log" on page 37 "How to Configure UCMDB Log Levels" on page 38
	showAllBinariesApplied	"How to View and Track Hotfixes Applied on UCMDB Server" on page 46
	viewSystemInformation	"How to View and Track Hotfixes Applied on UCMDB Server" on page 46
Settings Services	getSettingDefaultValue	"UCMDB Browser JMX Methods" on page 50
	listSensitiveSettings markSettingAsNonsensitive markSettingAsSensitive	"How to Mark Sensitive Settings and Enable Storing Encrypted Data in the Database Using JMX" on page 148
	setGlobalSettingValue	"How to Enable Validation of the Host Header of a Request" on page 44
	getInternalSetting setInternalSetting	"How to Increase the Number of Threads for Data Push Jobs" on page 88
	setSettingValue	"How to Use the User Activity Log" on page 37

Service	Method	Link to document
		"UCMDB Browser JMX Methods" on page 50
		"How to Enable CI Type Tenant Owner Verification during the Matching Phase of Identification" on page 92
		"How to Enable Attribute Name Verification during the Matching Phase of Identification" on page 92
	showSettingsByCategory	"How to Use the User Activity Log" on page 37
Supportability Services	listSupportCategories	"How to Access Support Using the JMX Console" on page 28
	runSupportHandlersForAllCategories	"How to Access Support Using the JMX Console" on page 28
	runSupportHandlersForSpecificCategories	"How to Access Support Using the JMX Console" on page 28
	selectAndRunSupportHandlers	"How to Access Support Using the JMX Console" on page 28
Topology Search Services	debugSolrQuery	"UCMDB Browser JMX Methods" on page 50
	restoreFactoryDefaults	"UCMDB Browser JMX Methods" on page 50
TQL Services	calculateTqlAdHoc	"How to Define and View a Layout Selection for a TQL Query" on page 60
	exportTQL	"Web Service API - executeTopologyQueryWithParameters " on page 100
UCMDB Integration	getEncryptedPasswordForURL	"How to Encrypt the Password of a Direct Link" on page 61
	setCMDBSuperIntegrationUser	"How to Create an Integration User" on page 97
UI Server frontend settings	setUseFrontendURLBySettings	"How to Configure a Reverse Proxy" on page 111
		"How to Set the IIS server as the Front-End Server for UCMDB" on

Service	Method	Link to document
		page 146
	showFrontendURLInSettings	"How to Configure a Reverse Proxy" on page 111
URM Services	listResources	"How to Increase the Number of Threads for Data Push Jobs"
	listResourceTypes	"UCMDB Browser JMX Methods" on page 50 "How to View Discovery Resource History" on page 75 "How to Enable CI Type Tenant Owner Verification during the Matching Phase of Identification" on page 92

Data Flow Management JMX Methods

Service	Method	Link to document
CMClient	displayCacheConfiguration	"How to Configure the Confidential Manager Client's Cache Encryption Settings on the Probe" on page 122
	isCMClientInitialized	"How to Check the Confidential Manager Connection" on page 88
	setCacheEncryption Algorithm	"How to Configure the Confidential Manager Client's Cache Encryption Settings on the Probe" on page 122
	setCacheEncryptionLibrary	"How to Configure the Confidential Manager Client's Cache Encryption Settings on the Probe" on page 122
	setCacheInitString	"How to Configure the Confidential Manager Client's Cache Encryption Settings on the Probe" on page 122
	setCacheMacDetails	"How to Configure the Confidential Manager Client's Cache Encryption Settings on the Probe" on page 122
	setLWSSOInitString	"How to Configure Confidential Manager Client Authentication and

Service	Method	Link to document
		Encryption Settings on the Probe" on page 121
	setTransportEncryptionAlgorithm	"How to Configure Confidential Manager Communication Encryption on the Probe" on page 121
	setTransportEncryptionLibrary	"How to Configure Confidential Manager Communication Encryption on the Probe" on page 121
	setTransportInitString	"How to Configure Confidential Manager Communication Encryption on the Probe" on page 121
	setTransportMacDetails	"How to Configure Confidential Manager Communication Encryption on the Probe" on page 121
GeneralUtils	executeLogGrabber	"Data Flow Probe Log Files" on page 84
	showAllBinariesApplied	"How to View and Track Hotfixes Applied on Probe" on page 80
JobsInformation	activateJob	"How to View Job Information on the Data Flow Probe" on page 65
	activateJobOnDestination	"How to View Job Information on the Data Flow Probe" on page 65
	start/stop	"How to View Job Information on the Data Flow Probe" on page 65
	viewJobErrorsSummary	"How to View Job Information on the Data Flow Probe" on page 65
	viewJobExecHistory	"How to View Job Information on the Data Flow Probe" on page 65
	viewJobProblems	"How to View Job Information on the Data Flow Probe" on page 65
	viewJobResultCiInstances	"How to View Job Information on the Data Flow Probe" on page 65
	viewJobResults	"How to View Job Information on the Data Flow Probe" on page 65
	viewJobsStatuses	"How to View Job Information on the Data Flow Probe" on page 65

Service	Method	Link to document
	viewJobStatus	"How to View Job Information on the Data Flow Probe" on page 65
	viewJobTriggeredCIs	"How to View Job Information on the Data Flow Probe" on page 65
	viewJobTriggeredCIsWithErrorId	"How to View Job Information on the Data Flow Probe" on page 65
MainProbe	dropUnsentResults	"How to Delete Unsent Probe Results " on page 81
	getEncryptedDBPassword	"How to Modify the PostgreSQL Database Encrypted Password" on page 115
	getEncryptedKeyPassword	"How to Set the JMX Console Encrypted Password" on page 116 "How to Set the UpLoadScanFile Password" on page 117 "How to Encrypt the Probe Keystore and Truststore Passwords" on page 139
	setSharedKey	"How to Set Shared Key for Encrypting or Decrypting the InfrastructureSettings.xml File Using JMX" on page 150
NormalizationRuleBase	scanForScanFileRules	"How to View Discovery Rules" on page 74
	scanForSNMPRules	"How to View Discovery Rules" on page 74
	viewNormalizationRuleById	"How to View Discovery Rules" on page 74
	viewNormalizationRuleByNiceId	"How to View Discovery Rules" on page 74
	viewNormalizationRules	"How to View Discovery Rules" on page 74
SecurityManagerService	importEncryptionKey	"How to Generate or Update the Encryption Key for Confidential Manager" on page 125
XmlEnricherMonitor	viewXmlEnricherStatuses	"How to Check XML Enricher Health Using JMX" on page 87

Configuration Manager JMX Methods

Service	Method	Link to document
ImportExport Service	activateAutoManageResource	"Configuration Manager JMX Methods" on page 101
	exportData	"Configuration Manager JMX Methods" on page 101
	exportPolicies	"Configuration Manager JMX Methods" on page 101
	exportViews	"Configuration Manager JMX Methods" on page 101
	importData	"Configuration Manager JMX Methods" on page 101
	listAllPolicies	"Configuration Manager JMX Methods" on page 101
	listAllViews	"Configuration Manager JMX Methods" on page 101
Licensed Content Service	deactivateAutomanagedResources	"Configuration Manager JMX Methods" on page 101
View Service	supportLargeViews	"Configuration Manager JMX Methods" on page 101
	updateFoldingRules	"Configuration Manager JMX Methods" on page 101

Chapter 2: Administration Methods

This chapter includes:

Unified Resource Manager (URM) JMX Methods

The Unified Resource Manager (URM) is an XML-based repository for CMDB resources. A resource is defined as all CMDB data other than CIs. Examples of resources include TQL queries, views, users, and the class model, as well as discovery resources such as discovery scripts, integration and discovery adapters, discovery jobs, and so on.

The URM can be accessed using the JMX console only. From the JMX console page, click **UCMDB:service=URM Services** to open the JMX page with the relevant methods.

For more information, see How to View Discovery Resource History in the *HPE Universal CMDB Data Flow Management Guide*.

> **Caution:** Never change a resource from the URM.

How to Manage UCMDB Licenses Using the JMX Console

You can manage the product licenses from the JMX Console. This task describes how to install a license.

1. On the UCMDB server, launch the Web browser and enter the following address:
   https://localhost:8443/jmx-console

   You may have to log in with a user name and password.

2. Click **UCMDB:service=Licensing Services** to open the JMX MBEAN View page.

3. Locate the **addLicense** method.

4. Enter your customer ID and the license key.

5. Click **Invoke**.

There are additional JMX methods available on the same page for the following functions:

- Installing a license from a file

- Displaying all active licenses

- Displaying all licenses (including expired licenses)

- Displaying a summary of active licenses

- Removing all licenses

How to Enable Remote Access to the JMX Console

Starting from version 10.30, the JMX console is secured by restricting access to it to server localhost only. Also, the JMX Console is no longer accessible through HTTP protocol, even if it is specifically re-

enabled.

Administrators attempting to access the JMX console from a remote machine may encounter all remote access attempts being redirected with no error.

The following message will be shown: "`Please wait, you will be redirected in a moment.`"

However, you can still enable remote access to JMX console when necessary.

To enable remote access to the JMX console,

> **Note:** This configuration affects both the UCMDB server JMX console and the probe JMX console.

1. On the UCMDB server, launch the Web browser and enter the following address:
   https://localhost:8443/jmx-console

   You may have to log in with a user name and password.

2. Locate **UCMDB:service=Settings Services** and click the link to jump to the Operations table.

3. Locate the **setGlobalSettingValue** operation.

4. Provide the following parameter values:

   name: restrict.jmx.to.localhost

   value: false

   > **Note:** Starting from UCMDB version 10.30, the default value is **true**, which disables remote access to the JMX console, you can only access the JMX console from localhost.

5. Click **Invoke**.

6. Restart the UCMDB server.

   The change takes effect and affects both the UCMDB server JMX console and the probe JMX console.

How to Download a Zip File of Log Files and Thread Dumps

You can produce a zip file that includes all logs and thread dumps. You create the file either through a JMX operation on the client machine, or by running a batch file on the UCMDB Server.

Thread dumps are created periodically: Once a minute, a thread dump snapshot is taken and is saved to a new file in the **C:\hp\UCMDB\UCMDBServer\runtime\log\threadDumps** folder. Thread dump files from the last hour are kept. This folder also holds the ad hoc Server snapshots that are generated during the **logGrabber** execution.

To generate the zip file from the client machine:

1. Launch the Web browser and enter the server address, as follows: **https://<UCMDB Server Host Name or IP>:8443/jmx-console**.

   You may have to log in with a user name and password.

2. Under **UCMDB**, click **UCMDB:service=Server services** to open the JMX MBEAN View page.

3. Locate the **executeLogGrabber** operation.

4. Click **Invoke**.

   A Server snapshot file with the name **LogGrabber_serverSnapshot_**
   <current date and time>.txt is created in the following location:
   C:\hp\UCMDB\UCMDBServer\runtime\log\threadDumps. This is a thread dump that includes the threads of the Server framework only.

5. In the File Download dialog box, you can open the **logGrabber_<current time>.zip** file, or download it to the client machine.

To generate the zip file from the UCMDB Server:

1. Access the following folder on the UCMDB Server:
   C:\hp\UCMDB\UCMDBServer\tools\logGrabber.

2. Run the **logGrabber.bat** file.

   The **LogGrabber_<current time>.zip** file is created in the following location:
   C:\hp\UCMDB\UCMDBServer\runtime. This is a thread dump that includes the threads of the Server framework only.

How to Retrieve UCMDB Server Logs for a Specific Time Frame

You can produce a zip file containing all UCMDB server logs for a given time frame. This is intended for support engineers or other users who need to obtain logs for a specific time frame.

To generate the zip file from the client machine:

1. Launch the Web browser and enter the server address, as follows: **https://<UCMDB Server Host Name or IP>:8443/jmx-console**.

   You may have to log in with a user name and password.

2. Under **UCMDB**, click **UCMDB:service=Server Services** to open the JMX MBEAN View page.

3. Locate the **executeServerLogParser** operation.

4. Enter the start time in the required format.

5. (Optional) Enter an end time. If you do not provide an end time, the current time that the JMX method is invoked is used.

6. Click **Invoke**.

   When the process has finished, the file can be downloaded from the browser.

Limitations

- The zip file is also located on the UCMDB server machine as the **c:\hp\UCMDB\UCMDBServer\runtime\ParsedLogGrabber_<time>.zip** file. For maintenance purposes, this file must be manually deleted.

- The folder **c:\hp\UCMDB\UCMDBServer\runtime\log\ParsedLogs_<date>** is also created and contains the unzipped contents. For maintenance purposes, this file must be manually deleted.

- In high availability UCMDB deployments, this JMX method is running against one server only.

- Only logs from the same date can be parsed.

How to Access Support Using the JMX Console

Universal CMDB provides Supportability JMX methods to help HPE Software Support diagnose problems in your system. The methods use handlers for each category, which gather information relevant to that category from your system. When you run a handler for a category, it downloads a zip file of the information gathered for that category. Generally, HPE Software Support runs the Supportability methods to help provide a solution for the issue raised.

To access the Supportability methods:

1. On the UCMDB server, launch the Web browser and enter the following address:
   https://localhost:8443/jmx-console

   You may have to log in with a user name and password.

2. Click **UCMDB:service=Supportability Services** to open the JMX MBEAN View page.

3. The **listSupportCategories** method displays all the support categories:

   ○ To run all the handlers, invoke the **runSupportHandlersForAllCategories** method.

   ○ To run specific handlers, invoke the **selectAndRunSupportHandlers** method and select the handlers you want to run.

   ○ Alternatively, you can run specific handlers using the **runSupportHandlersForSpecificCategories** method. In the **categories** field, enter all the required handlers separated by commas, and click **Invoke**.

Supportability Handlers

The following handlers are available:

- **TQL.** Records the following data in the **TQL.properties** file:

  ○ Number of TQL queries

  ○ Number of active TQL queries

  ○ Number of active persistent TQL queries

  ○ Number of non-active TQL queries

  ○ It also creates the **Failed TQLs.txt** file, containing the list of failed active TQL queries

- **View.** Records the following data in the **View.properties** file:

  ○ Number of views

  ○ Number of views with a hierarchy definition

  ○ Number of views with a rule based hierarchy definition

  ○ Number of template based views

  ○ Number of perspective based views

  ○ Number of templates

- ○ Number of perspectives

- ○ Number of views of unknown type (this value should always be 0)

- **ViewArchive.** Records the following data in the **ViewArchive.properties** file:

- ○ Total number of archives

- ○ Total number of views with archives

- **Snapshots.** Records the following data in the **Snapshots.properties** file:

- ○ Total number of snapshots

- **Modeling.** Records the following data in the **Modeling.properties** file:

- ○ Number of business CIs

- ○ Number of models with content (models containing CIs)

- ○ Number of pattern based models

- ○ Number of instance based models

- **Enrichment.** Records the following data in the **Enrichment.properties** file:

- ○ Number of Enrichment rules

- ○ Number of all active Enrichment rules

- ○ Number of non-active Enrichment rules

- ○ Number of Enrichment business views

- ○ Number of all active Enrichment business views

- ○ Number of non-active Enrichment business views

- **High Availability.** Gathers the High Availability information from all of the servers in the cluster:

- ○ The High Availability cluster information is recorded in **HA.properties**:

  - **Is_ha_enabled**

  - Cluster name (if high availability is enabled)

  - Cluster nodes number (if high availability is enabled)

  - Cluster nodes names (if high availability is enabled)

- ○ The values for the High Availability settings (starting with **ha.**) are recorded in **HA settings.properties**

- **Domains.** Gathers IP range information and records in the **DomainsConfiguration Customer <CustomerID>.xml** file.

- **Management Zones.** Gathers rank, name, range definition, discovery activities, activity jobs, and scheduling information for management zones. Records this information in the **MngZonesConfiguration <CustomerID>.xml** file.

- **URM Counters.** Records each of the registered URM types and the number of instances of each one in the **Basic URM Counters.properties** file.

- **Settings.** Records the infrastructure settings and their values for this customer in the **Settings <customer ID>.properties** file.

- **Changed Settings.** Records the changed infrastructure settings and their values for this customer in the **Changed_Settings_<customer ID>.properties** file.

- **Memory and Thread Count Info.** Records the UCMDB memory usage and thread count in the **MemoryAndThreadInfo.html** file. The information is displayed in color: green, orange, and red. If the color is not green, it requires attention.

  > **Note:** LDAP authentication can generate many idle threads. Until these threads are closed, it can temporarily lead to a high thread number which may cause performance issues.

- **Authorization.** Records all the roles, users, user groups and role assignments in the **Authorization.properties** file. In a multi-tenancy environment, it records the tenant association of each role assignment.

- **Basic History.** Records the last date that the baseline process ran for each CI type in the **Basic History.properties** file.

- **History.** Records the number of history events in the current history table for each CI type in the **History.properties** file (only for CI types with history events)

- **Class Model.** Records the class model as an XML file, **Class Model.xml**. In a multi-customer environment, it records the number of different class models and their differences at the SDK level in the **Class Model.properties** file. (In a single-customer environment, this file contains only the information for the single customer.)

- **Model Update.** Records the following data in the **Basic Model Update.properties** file:

  ○ Number of CIs per CI type (only for CIs with instances)

  ○ Number of CIs connected to a **Node** CI type or one of its descendants

- **Reconciliation.** Records data processing statistics read from the **cmdb.reconciliation.log** file. It records the data in Excel format, in the **DiscoveryProcessingStatistics.xlsx** file. Data that might

require attention is highlighted in yellow and red. These thresholds are selected based on data collected for UCMDB version 10.xx medium and large deployments.

The following Excel sheets are created:

○ **dailyRates.** Records daily statistics. The following items are explained:

- **Total Model time.** Time needed to insert, update, or delete CIs to DB. A higher value might show a DB slowness and require DBA involvement.

- **Total Identify time.** Time required in building TQL queries, calculating TQL queries, matching between bulk and what was brought with TQL queries from the UCMDB server.

- **Total DataIN time.** Time required for merge operations and recursive merge operations.

- **Daily Usage(% from 24h).** Amount of time in a day the server is busy with data in processing. If this field is highlighted in yellow or red, the server might be overloaded or there is a performance slowness.

○ **Jobs Throughput Rates Detailed**. Data recorded per discovery/integration job from the time the data is analysed. (First timestamp recorded in the logs.)

○ **dailyRatesPerJob.** Data recorded per discovery/integration job from the time the data is analyzed but breakdown per day.

○ **dailyRatesPerDataSource.** Data recorded per source type and display by day.

○ **FailedBulksInfo.** Number of bulks that failed to be recorded by the discovery job. It shows the number of bulks that failed and the time spent. At the end a summary is made for all failed bulks.

○ **RemoveByIDStats.** Records data for the CIs that are marked by the probe as candidates for deletion or CIs to be deleted.

○ **dailyRatesPerProbe.** Records data per day for each probe that reports data. Also it fetches all data that are not created through probe as 'Not through Probe'.

○ **Jobs Throughput Rates Per Probe.** Records data reported by every probe from the time the data is analyzed. (First timestamp recorded in the logs.)

○ **General Info.** Shows the timeframe analyzed and recorded in the Excel file. Also records time spent for successful and failed processed bulks . **Number of data in bulks ignored from log file** shows the number of bulks that were ignored. Possible reasons are that data was not properly read or that the reconciliation bulk was not logged properly in the log (every bulk must have two lines: entry line and summary line logged in consecutive rows in the log file).

Note:

○ The data is read from the **cmdb.reconciliation.log** file based on regex. If the log4j layout is changed for the **cmdb.reconciliation.log** file, the data cannot be parsed.

○ A job might show a performance degradation if the total time is higher than 600 seconds.

- **Data In.** Records actual deletion period and deletion candidate period information of the root CI type that was overwritten by the settings for child CI types in the **Data In.properties** file. It also checks for inconsistency in the database (objects or links that exist in the root CIT's table but not in the subtype's table or the other way around). The inconsistent objects are recorded in the **inconsistencyInModel.txt** file and the inconsistent links are recorded in the **inconsistencyLinks.txt** file.

How to Set Master Keys

You can use the JMX console to change the master key that is used to encrypt all UCMDB keys.

Change the master key for a cluster

This method assumes that your UCMDB environment is deployed in a high-availability setup.

Caution:

- This method involves a restart of the entire cluster, so plan accordingly. It is recommended to change the master key of the cluster when there is little or no load on the servers. For example, you should avoid using this method during data-in operations.

- Do not change any settings in the time period between changing the master key and restarting the server. Not following this instruction may result in a failure to start the server.

- Machines that are not up or that will be added later to the cluster will need to be configured manually. Until they are configured, at most they can run as reader machines; trying to run them as writer machines will fail.

1. Back up the **c:\hp\UCMDB\UCMDBServer\conf\cmdb.conf** file and the values for the following settings:

   ○ ha.cluster.authentication.keystore.password

   ○ ha.cluster.authentication.shared.secret

   ○ ha.cluster.message.encryption.keystore.password

- ○ ssl.server.keystore.password

- ○ ssl.server.truststore.password

2. Make sure all the servers in the cluster are up and running.

3. On the writer machine, launch the Web browser and enter the following address to log in to the JMX console: **https://localhost:8443/jmx-console**.

   > **Note:** If a load balancer is present, you must bypass it and not log on to the writer machine through a load balancer.

4. Do one of the following:

   - ○ Search for **changeMasterKeyForCluster**.

   - ○ Click **UCMDB:service=Security Services > changeMasterKeyForCluster**.

5. Enter and confirm the master key, and click **Invoke**. The master key will be changed first on the writer machine and then on all reader machines.

   > **Note:**
   >
   > The master key must be exactly 32 characters and include at least one of each of the following four types of characters:
   >
   > - ○ Uppercase alphabetic characters
   >
   > - ○ Lowercase alphabetic characters
   >
   > - ○ Numeric characters
   >
   > - ○ At least one of these special characters: **,\:/._?&%=+-[]()|**

6. Restart all the machines in the cluster. You can use the JMX method **High Availability Services > restartCluster** to do this.

   > **Note:** Restart the cluster immediately after changing the master key. If you do not, future database connections may fail.

Change the master key for a new machine in a cluster

If at least one of the following settings was changed, use Method A; otherwise, use Method B:

- ● ha.cluster.authentication.keystore.password

- ● ha.cluster.authentication.shared.secret

- ha.cluster.message.encryption.keystore.password

- ssl.server.keystore.password

- ssl.server.truststore.password

Method A

This method assumes that you already have properly configured a master key for the writer machine that is up and running in the cluster. If not, follow the instructions in "Change the master key for a cluster" on page 33.

1. Copy the **c:\hp\UCMDB\UCMDBServer\bin\wrapper.conf** file from the writer machine to the same location on the new (reader) machine.

2. Restart the server.

Method B

1. Back up the **c:\hp\UCMDB\UCMDBServer\conf\cmdb.conf** file.

2. On the writer machine, launch the Web browser and enter the following address to log in to the JMX console: **https://localhost:8443/jmx-console**.

3. Do one of the following:

   ○ Search for **changeMasterKey**.

   ○ Click **UCMDB:service=Security Services > changeMasterKey**.

4. Enter and confirm the master key, and click **Invoke**.

   Note: The master key must be exactly 32 characters and include at least one of each of the following four types of characters:

   ○ Uppercase alphabetic characters

   ○ Lowercase alphabetic characters

   ○ Numeric characters

   ○ At least one of these special characters: **,\:/._?&%=+-[]()|**

5. Restart the machine.

   Note: Restart the cluster immediately after changing the master key. If you do not, future database connections may fail.

Revert the master key for a cluster to its default value

This procedure resets the master key for an entire cluster.

1. Make sure all the servers in the cluster are up and running.

2. On the writer machine, launch the Web browser and enter the following address to log in to the JMX console: **https://localhost:8443/jmx-console**.

   > **Note:** If a load balancer is present, you must bypass it and not log on to the writer machine through a load balancer.

3. Do one of the following:

   ○ Search for **restoreMasterKeyForCluster**.

   ○ Click **UCMDB:service=Security Services > restoreMasterKeyForCluster**.

4. Click **Invoke**. The master key will be changed first on the writer machine and then on all reader machines.

5. Restart all the machines in the cluster. You can use the JMX method **High Availability Services > restartCluster** to do this.

   > **Note:** Restart the cluster immediately after changing the master key. If you do not, future database connections may fail.

Revert the master key for a machine that was down when master key was reverted for whole cluster

1. Back up the **c:\hp\UCMDB\UCMDBServer\conf\cmdb.conf** file.

2. On the writer machine, launch the Web browser and enter the following address to log in to the JMX console: **https://localhost:8443/jmx-console**.

3. Do one of the following:

   ○ Search for **restoreMasterKey**.

   ○ Click **UCMDB:service=Security Services > restoreMasterKey**.

4. Click **Invoke**.

5. Restart the machine.

> **Note:** Restart the cluster immediately after changing the master key. If you do not, future database connections may fail.

How to Use the User Activity Log

When troubleshooting a problem in your system, another useful tool is the User Activity log. When activated, this log records all the actions performed on your system, enabling HP Software Support to reproduce the problem and troubleshoot it.

To activate the User Activity log, first verify that it is enabled:

1. On the UCMDB server, launch the Web browser and enter the following address:
   https://localhost:8443/jmx-console

   You may have to log in with a user name and password.

2. Click **UCMDB:service=Settings Services** to open the JMX MBEAN View page.

3. Locate the **showSettingsByCategory** method.

4. Enter General Settings as the category name and click **Invoke**.

5. Locate the **mam.web.user.activity.log.enabled** setting and verify that it is set to **true**.

6. If it is set to false, go back to the **Settings Services** page, and select the **setSettingValue** method.

7. Enter **mam.web.user.activity.log.enabled** as the setting and **true** as the value and click **Invoke**.

Next, change the log level to INFO:

1. In the JMX Console, click **UCMDB:service=Server Services**

2. Locate the **loggersLevels** method and click **Invoke**.

3. Locate the **com.hp.ucmdb.uiserver.aspects** logger and select **INFO** from the drop-down list.

4. Click **Update loggers**.

The log is now activated. Perform the actions that led to the problem. The User Activity log automatically records them.

When you are finished, disable the log using the **loggersLevels** method and selecting **ERROR** as the level for the **com.hp.ucmdb.uiserver.aspects** logger.

How to Configure UCMDB Log Levels

This task describes how to specify the log level for UCMDB log files.

1. On the UCMDB server, launch the Web browser and enter the following address:
   https://localhost:8443/jmx-console

   You may have to log in with a user name and password.

2. Click **UCMDB:service=Server Services** to open the JMX MBEAN View page.

3. Locate the **loggersLevels** method.

4. Click **Invoke**.

5. From the list next to each log file name for which you want to set the level, select the required log level (OFF, FATAL, ERROR, WARN, INFO, DEBUG, TRACE, or ALL).

6. Click **Update loggers**.

How to Check the Database Connection

To check that the database server is up and running:

1. Launch the Web browser and navigate to: **https://<Server name>:8443/jmx-console**, where **<Server name>** is the name of the machine on which Universal CMDB is installed.

2. Under **UCMDB**, click **UCMDB:service=Dal Services** to open the JMX MBean View.

3. Invoke the function **getDbContext** with a **customerID** parameter value of **1**.

4. Check that the operation result shows no problems.

How to View the KPI Dashboard

You can use the key performance indicator (KPI) dashboard in the JMX console to view the following information of the UCMDB Server:

- Database connection summary

- Discovery processing statistics

- Operations information

Follow these steps to view the KPI dashboard:

1. Launch a web browser and enter the following address: **https://localhost:8443/jmx-console**.

   You may have to log in with a user name and password.

2. Locate **KPI:service=Dashboard** and click the link to open the JMX MBEAN View page.

3. Invoke one of the following methods:

   ○ **viewDbConnectionSummary**: displays database connection summary

   ○ **viewDiscoveryProcessingStatistics**: displays discovery processing statistics

   ○ **viewOperationsInformation**: displays operations information

   Note: The **showHTML** option controls if the information is displayed in the HTML or JSON format.

   True: the information is displayed in the HTML format

   False: the information is displayed in the JSON format

If the database is Oracle, the **viewDbConnectionSummary** method can gather and display database sessions. To enable this capability, the customer DBA must grant rights to the UCMDB:

- grant select on v_$session to *<db_schema>*;

- grant select on v_$locked_object to *<db_schema>*;

- grant select on v_$sqlarea to *<db_schema>*;

- grant select on v_$transaction to *<db_schema>*;

The UCMDB Server regularly exports the information displayed in the KPI dashboard into the following folder: **UCMDBServer/runtime/log/statistics/metrics_json**. The automatic export is performed by a scheduled job: **KPI Dashboard Dump**, which by default runs hourly. The exported files have names that resemble the following examples:

- DbConnectionSummary2016.05.01-11.30.00.json

- DiscoveryProcessingStats2016.05.01-11.30.00.json

If you want to start or stop this automatic export, use one of the following options:

- Activate or deactivate the **KPI Dashboard Dump** job from **Administration > Scheduler**

- Set the **statistics.gathering.enabled** setting to true (activate) or false (deactivate).

Performance Statistics Dashboard

The UCMDB Server automatically captures performance metrics and saves the data statistically in .csv files under the **UCMDBServer\runtime\log\statistics\metrics** folder.

To define the interval (measured in minutes) at which the UCMDB Server gathers performance metrics:

1. In UCMDB, go to **Administration > infrastructure Settings Manager > General Settings**.

2. Locate **Statistics gathering interval** setting and specify a time interval in minutes. The default value is 15 minutes.

3. Restart the UCMDB Server.

The following table explains each column in the .csv files.

Column	Information
t	Timestamp (epoch)
count	Number of invocations [1]
max	Max invocation duration [1]
mean	Average invocation duration [1]
min	Min invocation duration [1]
stddev	Standard Deviation of the values
p50	50% quantile [1,2]
p75	75% quantile [1,2]
p95	95% quantile [1,2]
p98	98% quantile [1,2]
p99	99% quantile [1,2]
p999	99.9% quantile [1,2]
mean_rate	Average event rate (per second)
m1_rate	Average event rate in the last minutes
m5_rate	Average event rate in the last 5 minutes
m15_rate	Average event rate in the last 15 minutes

Column	Information
rate_unit	Unit for the rate (for example, calls/second)
duration_unit	Unit for the durations (for example, seconds)

> **Note:**
>
> 1. All these values are collected since the last metric reset. Every 15 minutes after dumping the statistics to a CSV, the metrics (counters, durations) are reset to 0.
>
> 2. A quantile is calculated by taking the entire data set, sorting it, and taking the value in the middle (or 1% from the end, for the 99th percentile). For example, the 75% quantile shows the median of the lower 75% values. These quantiles apply to the set of durations of each invocation.
>
> Counters are not reset every 15 minutes, so their values keep growing and show overall counts since the system startup.

Captured Metrics

The following tables show the captured metrics.

Data-in statistics

Metrics	Measurement	Note
data-in.<submodule>	Timer	A breakdown of times in the data in process, by submodule (and submodules of submodules)
data-in.total	Timer	An overall timer from start to end for the data-in operation

UCMDB API Request statistics

Metrics	Measurement	Note
api-executed.<Operation>	Timer	A timer over API requests
api-failed	Counter	A counter that shows the number of failed API requests

Server Session statistics

Metrics	Measurement
server-sessions	Counter

DB Connection statistics

Metrics	Measurement
db-connection-borrowed	Counter
db-connection-borrow-failed	Counter
db-connection-returned	Counter
db-connection-invalidated	Counter
Connection timers	Timer

DAL statistics

Metrics	Measurement	Note
dal-executed.<sql>	Timer	A timer over the execution of a specific, long-running SQL (>1 minute)
dal-executed	Timer	Overall timer across all DAL executions
dal-failed	Counter	
db-rows-updated	Counter	A counter showing how many rows all the SQLs modified

TQL statistics

Metrics	Measurement	Note
tql-calculation.<tql>	Timer	A timer over the execution of a specific, long-running TQL (>1 minute)
tql-calculation	Timer	Overall timer across all TQL executions
tql-results	Timer	Counter for the total amount of CIs returned by TQL queries so far

Startup statistics

Metrics	Measurement	Note
startup	Timer	A timer over the startup of UCMDB and shows the time needed for the server startup

Operation statistics

Metrics	Measurement	Note
operation-executed.<"sync"/"async">.<CallerApp>.<OperationName>	Timer	Timers over specific operations • "sync": the caller is waiting for the operation to finish • "async": the operation runs in an async thread
operation-executed."end-to-end".<"sync"/"async">.<CallerApp>.<OperationName>	Timer	Timers that are over end-to-end operations, meaning the operation in the timer name is the first operation in an operation call stack
operation-failed	Counter	A counter that shows the number of failed operations

UI Server statistics

Metrics	Measurement	Note
ui-service-executed.<Service>	Timer	Timers over specific UI Server service calls
ui-service-failed	Counter	

JVM statistics

Metrics
"gc"
"buffers"
"memory"
"threads"

Jetty statistics

Metrics
war-context-handler.<context>
root-handler
empty-root-handler
jetty-connections

Jetty Queued Thread Pool

Metrics
utilization
utilization-max
size
jobs

Statistics on overall counters (objects, links, merges) for data-in operations

Metrics	Measurement
merged-cis	Counter
type_changes	Counter
merge-operations	Counter
ignored-from-cmdb	Counter
ignored-from-bulk	Counter
updated-objects	Counter
updated-links	Counter

How to Enable Validation of the Host Header of a Request

To enable validation of host header of a request based on a regular expression,

1. Access the UCMDB server JMX console: Launch a Web browser and enter the following address: **https://<UCMDB machine name or IP address>:8443/jmx-console**. You may have to log in with a user name and password.

2. Locate **UCMDB:service=Settings Services** and click the link to jump to the Operations table.

3. Locate the **setGlobalSettingValue** operation.

4. To enable validation of host header of a request, provide values for the following parameters for the **setGlobalSettingValue** method:

- **name**: **security.filter.header.allowed.host**

- **value**: <enter a JAVA regex that matches a valid host>

  For example,

  - **mymachine**, will only accepts a host header that contains the value **mymachine**.

  - **mymachine.*|localhost**, will accept **mymachine.mydomain.com**, **mymachine.subd1.domain1.com** or any domain starting with **mymachine** or **localhost** as host.

  The default value of the **security.filter.header.allowed.host** setting is **.***. It affects URLs containing **/ucmdb-ui/** only. For example, **https://localhost:8443/ucmdb-ui/applet/applet.jsp**.

5. Click **Invoke**. The setting takes effect immediately.

6. Restart the whole UCMDB cluster to ensure that the setting is picked up by all readers.

How to Show or Hide the "Cannot invoke trigger" Error Message on UI

Version 10.21 introduced a new JMX setting **appilog.collectors.ShowCanNotInvokeError** in the **getInternalSetting** method, allowing you to show or hide the "Cannot invoke trigger" error message on UI. The default value for the setting is **true**.

To hide the "Cannot invoke trigger" error message,

1. Go to **JMX Console > UCMDB:service=Settings Services > getInternalSetting**.

2. Invoke the **getInternalSetting** method with the following parameters:

   - **customerID**: <Customer ID>

   - **key**: appilog.collectors.ShowCanNotInvokeError

3. On the result page,

   - To show the error message, replace the **null** value with **true** (default) and click **Set**.

   - To hide the error message, replace the **null** value with **false** and click **Set**.

   Note that for the error message displayed for a specific trigger, the change takes effective the next time the trigger CI is triggered. Since not all triggers are scheduled to run at the same time, you may still see occurrences of the error message on the UI for some time before it completely goes off.

How to View and Track Hotfixes Applied on UCMDB Server

Version 10.30 introduced the **showAllBinariesApplied** JMX method, allowing you to easily view and track all information about the hotfixes you deployed. Also, the **viewSystemInformation** operation is enhanced to include IDs of hotfixes applied.

To view and track hotfixes deployed on UCMDB server,

1. On the UCMDB server, launch the Web browser and enter the following address:
   https://localhost:8443/jmx-console

2. Enter the JMX console authentication credentials (Login name = **sysadmin**).

3. Locate the following operations in the **UCMDB:service=Server Services** category:

   ○ To view IDs of hotfixes applied only, locate the **viewSystemInformation** operation.

   ○ To view and track detailed information about hotfixes deployed, locate the **showAllBinariesApplied** operation.

4. Click **Invoke**.

   ○ The result page of the **viewSystemInformation** operation displays the system information of the UCMDB server, including the IDs of hotfixes deployed on the server. This is also saved in the supportability JMX general category output.

   ○ The result page of the **showAllBinariesApplied** operation displays detailed information about the hotfixes deployed, including hotfix type (server backend, UI server, or applet), readme file, if there're any overlapping files, and if there are any wrongly placed files. See the screenshot below for an example.

```
Mbean: UCMDB:service=Server Services. Method: showAllBinariesApplied

The following Server backend hotfixes/binaries are applied
ID: HF_QCCR1H12313              Readme File not available.
ID: HF_QCCR1H11244              PROBLEM DESCRIPTION Trace is activated on the callhome servlet port 80
ID: HF_QCCR1H23211             Readme File not available.

The following UI-Server hotfixes/binaries are applied
ID: HF_QCCR1H101112            Readme File not available.

The following UI-Applet hotfixes/binaries are applied
ID: HF_QCCR1H1121233           Readme File not available.

NOTE: The following hotfixes/binaries have duplicate files: double-check them with HP
ID: HF_QCCR1H11244              [\com\hp\ucmdb\history\dal\command\modify\HistDalComplementLayoutPerTableCommand.class, \com\hp\ucmdb\history\operation\update\HistComplementLayoutPerClass.class]
ID: HF_QCCR1H23211             [\com\hp\ucmdb\history\dal\command\modify\HistDalComplementLayoutPerTableCommand.class, \com\hp\ucmdb\history\operation\update\HistComplementLayoutPerClass.class]
```

High Availability Mode JMX Methods

Replace the Writer Server

In the JMX Console, you can invoke the **High Availability Services > suggestNewWriterServer** method where you can suggest which server (serverID) should replace the Writer server.

High Availability Cluster Authentication

To enable cluster authentication:

1. In UCMDB, go to **Administration > Infrastructure Settings Manager**.

2. Find the setting **Enable joining High Availability cluster authentication** and set it to **true**.

3. Provide a single server authentication keystore (certificate + private and public keys) in JKS format. This keystore will be placed on all the servers and used for authenticating when connecting to a high availability cluster.

   Place the keystore in the following location: **<UCMDB installation folder>\conf\security** and name it **cluster.authentication.keystore.**

   > **Note:** The UCMDB comes with this keystore pre-configured out-of-the-box. This keystore is the same for all clean UCMDB installations, and thus not secure. If you wish to securely authenticate join requests, delete this file and create a new one.

4. Generate a cluster authentication keystore as follows:

   a. From C:\hp\UCMDB\UCMDBServer\bin\jre\bin, run the following command:

   ```
   keytool -genkey -alias hpcert –keystore <UCMDB installation
   folder>\conf\security\cluster.authentication.keystore -keyalg RSA
   ```

   The console dialog box opens and asks you for a new keystore password.

   b. The default password is **hppass**. If you want to use a different password, update the server by running the following JMX method: **UCMDB:service=High Availability Services: changeClusterAuthenticationKeystorePassword**

   c. In the console dialog box, answer the question **What is your first and last name?** by entering the name of the cluster.

   d. Enter the other parameters according to your organization's details.

e. Enter a key password. The key password must be the same as the keystore password.

   A JKS keystore is created in **<UCMDB installation folder>\conf\security\cluster.authentication.keystore**

5. Replace the old **<UCMDB installation folder>\conf\security\cluster.authentication.keystore** on all the servers in the cluster with the new keystore.

6. Restart all the servers in the cluster.

Changing the Key in the key.bin

In a High Availability environment with several servers, change the **key** in the **key.bin** as follows:

1. Go to the writer machine in the JMX. You can choose any machine in the cluster and click on the **writer** link on the top of each page.

2. In the UCMDB section of the console, click **UCMDB:service=Discovery Manager**.

3. Change the key in one of the following ways:

    ◦ Click **changeEncryptionKey** (this imports the existing encryption key)

    ◦ Click **generateEncryptionKey** (this generates a random encryption key)

4. On the writer machine, go to the file system and find the **key.bin** at: **C:\hp\UCMDB\UCMDBServer\conf\discovery\key.bin**

5. Copy the **key.bin** from the location on the writer machine to each one of other machines in the cluster to the folder: **C:\hp\UCMDB\UCMDBServer\conf\discovery\customer_1** and rename the destination file (for example, **key_new.bin**).

6. For each of the other servers (readers) do the following:

    a. Switch the reader to be a writer (you can do this from the High Availability JMX) and wait until it changes.

    b. Connect to the JMX of the current writer and click **UCMDB:service=Discovery Manager**.

    c. Click and invoke **changeEncryptionKey**, use the same details you entered in step 3 (for **newKeyFileName**, use the new name you assigned at step 5).

    d. Verify that you get the following message: **Key was created successfully**.

High Availability Cluster Message Encryption

Use cluster message encryption to encrypt all the messages in the cluster.

To enable cluster message encryption:

1. In UCMDB, go to **Administration > Infrastructure Settings Manager**.

2. Find the setting **Enable High Availability cluster communication encryption** and set it to **true**.

3. Provide a secret key for symmetric encryption on all the servers. The key should be placed in a keystore of type JCEKS in the following location **<UCMDB installation folder>\conf\security\cluster.encryption.keystore**.

   > **Note:** The UCMDB comes with this keystore pre-configured out of the box. This keystore is the same for all clean UCMDB installations, and thus not secure. If you wish to securely encrypt cluster messages, please delete this file, and create a new one by following this procedure.

4. From **<UCMDB installation folder>\bin\jre\bin**, run the following command:

   Keytool –genseckey –alias hpcert –keystore <UCMDB installation folder>\conf\security\cluster.encryption.keystore –storetype JCEKS

5. You will be asked for the new keystore password. The default password is "hppass". If you want to use a different password, you need to update the server by running the following JMX method:

   UCMDB:service=High Availability Services: changeClusterEncryptionKeystorePassword

6. Replace the old **<UCMDB installation folder>\conf\security\cluster.encryption.keystore** of all the servers in the cluster with this new keystore.

7. Restart the servers.

Troubleshooting - High Availability Mode

Upon every startup of the UCMDB server, the server sends a test message to the cluster to verify if it successfully connected to the cluster. If there is a problem with the connection, the message fails and the server is stopped to avoid the whole cluster getting stuck.

Some examples of wrong cluster encryption configuration are:

- Disabled encryption on one node when another node enabled it.

- Wrong or missing cluster.encryption.keystore

- Wrong or missing key in the keystore

If the server gets stuck because of a configuration issue, the error message is:

```
2012-09-11 17:48:23,584 [Thread-14] FATAL - #### Server failed to connect properly
to the cluster and its service is stopped! Please fix the problem and start it
again ####

2012-09-11 17:48:23,586 [Thread-14] FATAL -      Potential problems can be: wrong
security configuration (wrong or missing cluster.encryption.keystore, wrong key,
disabled encryption in a cluster with enabled encryption)
```

UCMDB Browser JMX Methods

How to Modify the Currently Indexed List

1. Go to **JMX Console > UCMDB:service=Topology Search Services**.

2. Choose one or more of the following operations:

   ○ editIndexerConfiguration – displays and enables editing of the **Search_Indexer_Configuration_XML** file.

   ○ editParserConfiguration – displays and enables editing of the **Search_Parser_Configuration_XML** file.

   ○ editRankingConfiguration – displays and enables editing of the **Search_Ranking_Configuration_XML** file.

3. For each operation, enter the relevant customer ID and click **Invoke**.

How to Enable/Disable the Search Engine

By default, the search engine is enabled (unless it was disabled during UCMDB installation).

To change the enable/disable setting:

1. Go to **JMX Console > UCMDB:service=Settings Services > setGlobalSettingValue**.

2. In the **name** field enter **cmdb.search.enabled**.

3. In the **value** field enter:

   true: If you want the search enabled.

   false: If you want the search disabled.

4. Click **Invoke**.

5. Restart the UCMDB server.

> **Note:** If you disable the Enhanced Search Engine, the UCMDB Browser automatically reverts to the legacy search engine.

How to Enable/Disable Searching for Federated Data

The search engine can be configured to perform searches on federated data. By default, it is disabled.

To enable or disable this setting:

1. Go to **JMX Console > UCMDB:service=Settings Services > setSettingValue**.
2. In the name field enter **cmdb.federation.search.enabled**.
3. In the value field enter:

   true: If you want to enable searching federated data.

   false: If you want to disable searching federated data.
4. Click **Invoke**.
5. Restart the UCMDB server.

How to Configure Repetition of the Enriching Mechanism

To configure the number of times that enriching is performed on search results:

1. Go to **JMX Console > UCMDB:service=Settings Services > setSettingValue**.
2. In the name field enter **cmdb.search.enriching.depth**.
3. In the value field enter the number of times that you want enriching to be repeated on search results.
4. Click **Invoke**.
5. Restart the UCMDB server.

How to Configure the Date Format

The search engine supports two dates formats: day-month-year (DMY) and month-day-year (MDY), which can be configured as follows:

1. Go to **JMX Console > UCMDB:service=Settings Services > setSettingValue.**

2. In the **name** field enter: **cmdb.search.date.format**.

3. In the **value** field enter the desired date format: **DMY**, **MDY**, or **both**.

4. Click **Invoke**.

5. Restart the UCMDB server.

How to Restore Factory Defaults

To restore the default configuration XML files from the factory content, go to **JMX Console > UCMDB:service=Topology Search Services** and invoke the **restoreFactoryDefaults()** method.

> **Caution:** This method overwrites the current configuration. You should back up the configuration files before invoking it.

Content of Solr Database

By default, the Solr search engine is embedded inside UCMDB server. To query it directly, go to **JMX Console > UCMDB:service=Topology Search Services** and invoke the **debugSolrQuery()** method.

How to Access the UCMDB Browser by IP Address

If you access the UCMDB Browser by IP address (not by FQDN), you should add the UCMDB Browser IP address to the UCMDB's trusted hosts. You can do this from the JMX console. Under **LW-SSO Configuration Management**, locate the **addTrustedIPs** method and invoke it using the UCMDB Browser IP address value.

How to Specify Persistency Values for Notifications

The length of time that notifications are retained and how often they are generated are defined in the JMX console:

1. Go to **JMX Console > UCMDB:service=Settings Services > setSettingValue**.

2. To change each setting, follow these steps:

    a. In the **name** field, enter one of the strings listed below:

        - tql.tracker.queue.evaluation.initial.delay.in.min - the initial delay (in minutes) after startup, before a TQL query is calculated.

        - tql.tracker.queue.evaluation.period.in.min - the interval (in minutes) of how often a TQL query is scheduled to run.

        - tql.tracker.queue.max.single.run.time.in.min - the maximum length of time (in minutes) for the system to work on calculating changes on CIs or TQL queries during a single execution.

        - tql.tracker.min.time.between.tracker.runs.in.min - the minimum length of time between two runs of a TQL query.

        Note: To find the default value for each setting, enter the required string in the **name** field of **getSettingDefaultValue** and click **Invoke**.

    b. In the **value** field, enter the value you want to set.

    c. Click **Invoke**.

3. Restart the UCMDB server.

How to Re-index the CIs of a Given CI Type

You can re-index the CIs of a given CI type from the CMDB model database for search purposes. To do this, follow these steps:

1. Go to **JMX Console** > **UCMDB:service=Topology Search Services** > **reindexCiType**.

2. In the **ciType** field, enter the CI Type that you want to re-index.

3. Select one of the following options for the **includeSubtypes** option:

    ○ **True**: Re-index all the subtypes of the specified CI Type.

    ○ **False**: Do not re-index the subtypes of the specified CI Type.

4. Click **Invoke**.

To check the status and progress of the re-indexing operation, use the **printStatusReport** JMX method. This method can display the information such as overall status, progress, and number of indexed entries.

Note:

- In a multi-UCMDB environment, the **reindexCiType** method triggers the re-indexing operation on all nodes in the cluster.

- During the execution of the re-index operation, search operations are allowed and will return results based on what is already indexed.

Package Manager JMX Methods

How to Deploy a Package

Follow these steps to deploy a package using the JMX console.

1. Launch your Web browser and enter the following address: **https://<server_name>:8443/jmx-console**, where **<server_name>** is the name of the machine on which Universal CMDB is installed.

2. Under **UCMDB**, click **UCMDB:service=Packaging Services** to open the JMX MBEAN View page.

3. Locate **deployPackages**.

4. In the **Value** box for the parameter **customerID**, enter the <customer id>.

5. In the **Value** box for the parameter **dir**, enter the name of the folder that contains the package's zip file. Ensure that you include the full path to the folder.

   Note: To deploy the package from the **basic_packages** directory, leave this box empty.

6. In the **Value** box for the parameter **packagesNames**, enter the name of the packages.

7. Select **True** to override job configurations changed in Universal Discovery.

8. Click **Invoke** to deploy the package.

How to View Package Deployment History

Each time you deploy packages, a report is created displaying the deployment status of those packages. Use the JMX console to view the deployment status report.

1. Launch the Web browser and navigate to: **http://<server_name>:8443/jmx-console**, where **<server_name>** is the name of the machine on which Universal CMDB is installed.

2. Under **UCMDB**, click **UCMDB:service=Packaging Services** to open the JMX MBEAN View page.

3. Locate **displayResourcesDeploymentHistory.**

4. In the **Value** box for the parameter **customerId**, enter the <customer id>.

5. In the **Value** box for the parameter **reportNum**, enter the number of the report you want to view.

6. Click **Invoke** to view the deployment status report of the packages.

How to Undeploy a Package

Follow these steps to undeploy a package using the JMX console.

1. Launch the Web browser and navigate to: **https://<server_name>:8443/jmx-console**, where **<server_name>** is the name of the machine on which Universal CMDB is installed.

2. Under **UCMDB**, click **UCMDB:service=Packaging Services** to open the JMX MBEAN View page.

3. Locate **undeployPackages**.

4. In the **Value** box for the parameter **customerId**, enter the <customer id>.

5. In the **Value** box for the parameter **packagesNames**, enter the name of the package you want to remove.

6. Click **Invoke** to undeploy the package.

How to Display Currently Deployed Packages

Follow these steps to display currently deployed packages using the JMX console.

1. Launch the Web browser and navigate to: **https://<server_name>:8443/jmx-console**, where **<server_name>** is the name of the machine on which Universal CMDB is installed.

2. Under **UCMDB**, click **UCMDB:service=Packaging Services** to open the JMX MBEAN View page.

3. Locate **displayDeployedPackages.**

4. In the **Value** box for the parameter **customerId**, enter the <customer id>.

5. In the **Value** box for the parameter **packagesNames**, specify the names of the packages you want to display.

6. Click **Invoke** to display the packages that are currently deployed.

How to Export Packages

Follow these steps to export resources from the CMDB to the server on which Universal CMDB is installed using the JMX console.

1. Launch the Web browser and navigate to: **https://<server_name>:8443/jmx-console**, where **<server_name>** is the name of the machine on which Universal CMDB is installed.

2. Under **UCMDB**, click **UCMDB:service=Packaging Services** to open the JMX MBEAN View page.

3. Locate **exportPackages**.

4. In the **Value** box for the parameter **customerId**, enter the <customer id>.

5. In the **Value** box for the parameter **packagesName**, enter the name of the package you want to export.

6. In the **Value** box for the parameter **outputDir**, enter the name of the folder on the Universal CMDB server to which you want to export the package's zip file. Ensure that you include the full path to the folder.

7. In the **Value** box for the parameter **userOnly**, select one of the following:

   ○ **True**. Export only the custom packages.

   ○ **False**. Export both custom and factory packages.

8. Click **Invoke** to export the package.

History DB Services JMX Methods

History DB Services JMX Method Name	Description
alignHistoryForType	Updates the history tables of a CI Type class according to the class description. This includes the main history tables, the removed CIs tables and the list attributes tables. If no CI type class is passed as a parameter all history will be aligned.
executeBaselineForClass	Executes the baseline process for a CI type class if needed.
executeFullBaseline	Executes the baseline process for all CI type classes.
findUnboundHistoryTables	Finds the history tables without appropriate history partition resource in URM. This method finds all the history tables from the database and compares them with the history tables from the history partitions URM resources. The method finds unbound tables for all the UCMDB Customers.
getHistoryChanges	Get history changes for an Object ID in a specified time frame.
getHistoryChangesCounterByType	Get number of changes according to class type and time frame.
getHistoryChangesExtendedCounter	Get an extended counter of changes for an Object ID and time frame.
getHistoryLayout	Get full history layout for Object ID from a specified date.
getRemovedData	Get removed data in a specified time frame.
initializeHistoryDBFromModel	Initialize History DB from Data Model. This method deletes all customer data from history tables, deletes URM History partitions, HDM and HDML tables for all history months, HDMR and HDMRL tables and HDM_ROOT table for the specified customer. After history is cleared the align process will recreate the history tables according to the class description and history will be initialized with the current data in the data model.
isHistoryEnabled	Determines whether history is enabled according to the settings.
purgeHistoryDBAccordingToInput	Purges the history DB according to the specified months to save back.

History DB Services JMX Method Name	Description
purgeHistoryDBAccordingToInput IncludingExtraMonths	Purges the history DB according to the input of how many months to save back and how many extra months to save old removed data.
purgeHistoryDBAccordingToSettings	Purges the history DB according to the purge settings.
purgeHistoryFailures	In order to easily troubleshoot history issues, last history failures for specific history operation(Baseline, Purge, Alignment etc) are saved inside URM. This JMX method clears the failures from the URM.
showHistoryTableForType	Shows the history tables for a specific CI type class.
showLastHistoryFailures	Displays a report of last UCMDB History failures for all history operations.
showLastHistoryFailuresForSpecific HistoryOperation	Displays a report of last UCMDB History failures for specific history operations: 1: BASELINE 2: BASELINE FROM DATA MODEL 3: PURGE 4: PURGE_MODEL_ REVISIONS 5: COMPLETE PARTIAL EVENTS 6: DELETE HISTORY TABLES/DATA FROM HISTORY ROOTS 7: ALIGNMENT
startHistoryDB	Starts the history database for saving CI history changes. Please note that this method will delete existing history and will recreate the history tables according to the data model.
stopHistoryDB	Stop the history database. This method will cause the existing CI history changes to be unusable.

History operations explained

1. BASELINE

   ○ **Baseline process:** Once a month a new history table is created for each CI Type, which will contain all the events that will be created in the next month. These Baseline events are added for each CI instance and will contain the whole CI data at the beginning of the month. This is needed to avoid unnecessary access to previous monthly tables.

   ○ **Baseline Process setting:**

   ```
   history.baseline.defined.start.date = 10 00
   ```

   Format: <day of month(1-28)><space><hour (00-23)>

   For example, by using the default setting, the April monthly table stores events occurred between April 10th at midnight through May 9th at 23:59:59.

Note:

- The starting time of the Baseline Process should not be the same as the History Purging starting time or Aging process start time.

- Because the Baseline process affects the population integration performance, make sure you schedule the Baseline Process to run at an appropriate distance from the discovery process time.

Chapter 3: Modeling Methods

This chapter includes:

How to Define and View a Layout Selection for a TQL Query

You can specify the attributes to include in the query results for each query node or relationship in a TQL query in the Element Layout tab of the Query Node Properties dialog box. Select the **Select attributes for layout** radio button and then select a CIT or relationship in the CIT pane. If you select **Specific Attributes** for the Attributes condition, only the attributes you move to the Specific Attributes pane are included in the query results for that element. If you select **All** for the Attributes condition, all of the available attributes are included in the query results for that element. In this case, you can select **Exclude specific attributes** and move selected attributes to the Excluded Attributes pane.

There is also an option to select attributes by qualifiers. If you select qualifiers in the **Attributes with the following qualifiers** field, all attributes that have the selected qualifiers are included in the query results for that element, in addition to the attributes selected in the Specific Attributes pane. In this case too, you can exclude selected attributes by moving them to the Excluded Attributes pane.

By default, the attribute settings you select for a CIT are automatically applied to its descendant CITs in the query results; however, the settings are not visible in the Element Layout tab of the dialog box. For example, if you select specific attributes to be included for the **Database** CIT, the same attributes are included for the **Oracle** CIT (a child CIT of **Database**), but if you select **Oracle** in the CIT pane, the Attributes condition displayed is **None** (the default condition).

You can then make an attributes condition selection for the child CITs themselves. If the parent CIT has **All** selected as the attributes condition, then the **Specific Attributes** option is disabled for the child CITs. If the parent CIT has **Specific Attributes** selected as the attributes condition, you can select **All** or **Specific Attributes** for the child CIT. If you select **Specific Attributes**, you can add more attributes

by moving them to the Specific Attributes pane. These are included in the query results along with the attributes inherited from the parent CIT's setting. Similarly, you can select attributes from the parent CIT's setting to exclude for the child CIT, by moving them to the Excluded Attributes pane. If the parent CIT has qualifiers selected to determine the attribute selection, these are also inherited by the child CIT. If you select additional qualifiers to filter the child CIT's attribute selection, the combined set of selected qualifiers is used to filter the attribute selection for the child CIT.

When you change the type of a query node or relationship using the Change Query Node/Relationship Type dialog box, the attributes selection for that element is lost.

If you import a package with a query that includes an attributes selection that is invalid for the selected query node, or if you make an invalid attributes selection using the JMX console, the query can be saved successfully and a warning appears in the log.

> **Note:** The layout selection is not visible in the query results in the user interface. To view the query results with the selected attributes, access the JMX console, select **TQL services**, and invoke the **calculateTqlAdHoc** method.

How to Encrypt the Password of a Direct Link

This task describes how to encrypt the password contained within a direct link using the JMX console.

To encrypt the password of a direct link using the JMX console:

1. Launch your Web browser and enter the following address: **https://<server_name>:<port number>/jmx-console**, where **<server_name>** is the name of the machine on which Universal CMDB is installed.

2. Under **UCMDB-UI**, locate **UCMDB Integration**.

3. Under **getEncryptedPasswordForURL**, enter your user name and the password to encrypt.

4. Click **Invoke** to view the encrypted string.

How to Rebuild the Database in Case of an Error

If an error occurs while working with views in the Modeling managers, when adding CIs to the CMDB, or when updating existing CIs, and the error log indicates that objects are missing in the database, do the following:

1. Perform a DB backup.

2. Access the JMX console and run the following methods under **service=DAL services**:

   ○ **rebuildModelViews**

   ○ **rebuildModelDBSchemaAndViews**

How to Rebuild Indexes on Microsoft SQL Databases

UCMDB is an online transaction processing (OLTP) application. It performs many insert, update, and delete operations each day, its indexes might become fragmented. The index fragmentation could be even higher if discovery jobs that run also modify the data.

To help rebuild fragmented indexes on Microsoft SQL database, the **RebuildIndexes** job is introduced. It uses a stored procedure to defragment the indexes. The **RebuildIndexes** job is deactivated by default. You can start the job manually when necessary.

How to manually start rebuilding fragmented indexes

You can start rebuilding fragmented indexes manually by using either of the following:

- Start the **RebuildIndexes** job from UCMDB UI

  a. On UCMDB UI, go to **Administration > Scheduler > Job Scheduler**.

  b. Select the **RebuildIndexes** job, and click **Edit** 🖉.

  c. Modify the job scheduler information in the Schedule section of the Job Definition dialog.

     It is recommended to run the job on a daily basis.

     For more information about the scheduler options, see "Job Definition Dialog Box" in the *HPE Universal CMDB Administration Guide*.

  d. Click **OK**.

- Invoke the **rebuildIndexes** JMX method from the JMX console

  a. Launch the Web browser and navigate to: **https://<Server name>:8443/jmx-console**, where **<Server name>** is the name of the machine on which Universal CMDB is installed.

  b. Under **UCMDB**, click **UCMDB:service=Dal Services** to open the JMX MBean View.

  c. Invoke the **rebuildIndexes** JMX method with a **customerID** parameter value of **1**.

  | **Note:**

- During the rebuilding of the indexes, the UCMDB database log file size may increase. Make sure that the UCMDB database log file has enough disk space.

- Before you invoke the **rebuildIndexes** JMX method, make sure that the discovery, enrichment, or other jobs that modify the data are not running.

- The **RebuildIndexes** job is visible only for Customer 1 (default customer) in UI, even if the Server has multiple customers.

- The **rebuildIndexes** JMX method recreates the indexes for all customers even if the customerID is set to 1, in case the server has multiple customers or has the authorized state.

How to modify the RebuildIndexes job definition and scheduler information

If necessary, you can modify the definition of the **RebuildIndexes** job. To do so,

1. On UCMDB UI, go to **Administration > Scheduler > Job Scheduler**.

2. Select the **RebuildIndexes** job, and click **Edit** .

3. Modify the job definition and scheduler information as necessary to meet your needs.

4. Click **OK**.

How to Modify Composite Indexes

A composite index contains multiple key columns. You can invoke the **modifyCompositeIndexes** JMX method to add or remove the **CMDB_ID** column as a key column in the indexes for the specified class. The method then modifies the indexes according to the specified parameters.

> **Note:**
>
> - This method only works for a Microsoft SQL or Oracle database.
>
> - When you perform a fresh install of the UCMDB 10.30 (or later), by default there are no composite indexes with the **CMDB_ID** as a key column on Oracle database (**ROOT** tables), and on Microsoft SQL databases (**ROOT** and **CDM** tables). To create the composite indexes with the **CMDB_ID** as a key column, follow the steps described in this procedure.

It is highly recommended that you follow these best practices for running the **modifyCompositeIndexes** method:

- Run the method when the probes are stopped and the database is not heavily used.

- Create a schema dump before running the method.

- For environments that execute heavy data-in operations, in order to speed up the INSERT statements, it is recommended to transform the database from composite index to non composite index by invoking the **modifyCompositeIndexes** method with **composite indexes** setting to **false**.

To modify composite indexes:

1. On the UCMDB server, launch the Web browser and enter the following address: **https://localhost:8443/jmx-console**

   You may have to log in with a user name and password.

2. Locate **UCMDB:service=Dal Services** and click the link to open the JMX MBean View, and then locate the **modifyCompositeIndexes** method.

3. Specify the following parameters of the method:

   - **customerId**: Specify the Customer ID (leave empty for the default customer)

   - **class name**: Specify the class name (allowed values are `root` and `all`)

   - **composite indexes**: Specify if the method rebuilds the indexes with **CMDB_ID** as a key column

     - `True`: **CMDB_ID** is a key column

     - `False`: **CMDB_ID** is not a key column

4. Click **Invoke**.

> **Note:** If any problem occurs, the operation rolls back and causes no change in the database.

How to Export the Class Model to XML

The Export to UML tool enables you to export selected sections of the UCMDB class model to a format compatible with UML tools, and to view the model as a UML diagram.

The input for the tool is the UCMDB class model XML file retrieved by the JMX service **UCMDB:service=Class Model Services/ exportClassModelToXml()**.

> **Note:** To access the JMX console, enter the following address in your browser: **http://<server_ name>:8443/jmx-console**, where **<server_name>** is the name of the machine on which Universal CMDB is installed.

Chapter 4: Data Flow Management Methods

This chapter includes:

How to View Job Information on the Data Flow Probe

This task describes how to view job information (for example, job threads and Trigger CIs) saved to the Data Flow Probe's PostgreSQL database. You work with the JMX console.

This task includes the following steps:

1. Access the MBean operations

   Use the following procedure to access the JMX console on the Data Flow Probe and to invoke the JMX operations.

   a. Launch the Web browser and enter the following address:

      http://<machine name or IP address>.<domain_name>:1977/

      where **<machine name or IP address>** is the machine on which the Data Flow Probe is installed. You may have to log in with the user name and password.

   b. Click the **Local_<machine name or IP address> > type=JobsInformation** link.

2. Locate the operation to invoke

   On the MBean View page, select **type=JobsInformation**. Locate the required operation.

3. Run the operation

   Click the **Invoke** button to run the operation. A message is displayed with the results of the operation run.

Reload	The number of seconds between automatic reloads of the JMX interface.
	0: The interface is never reloaded.
	Click the **Reload** button to manually reload the current page (if more operations have been added or removed).
Unregister	Do not touch (the view becomes inaccessible to the application that is running).

The following is a list of operations that can be invoked in the above procedure:

activateJob

Enter the name of a job and click the button to activate the job immediately. This operation returns a message, for example, **<job name> was triggered.**

> **Note:** The following message is displayed if the job has not been activated and there is no information about the job in the Probe's database:
>
> **Job '<job name>' does not exist in the Jobs Execution table (job was not activated!)**.

activateJobOnDestination

Enter the name of a job and a Trigger CI and click the button to activate the job immediately on a specific Trigger CI. This operation returns a message, for example, **The operation returned with the value: Job <job name> was triggered on destination <CI name>.**

> **Note:** Both the **JobID** and **triggerCI** fields are mandatory.

start/stop

These operations start and stop the **JobsInformation** service. Do not use these operations; instead, restart the Probe itself.

viewJobErrorsSummary

Enter the name of a job to return a list of error messages reported on this job, together with the error severity, the last time that the error was reported, and the number of Trigger CIs that have the error.

Click the entry in the **Number of Trigger CIs** column to view a list of one job's Trigger CIs with errors on the viewJobTriggeredCIsWithErrorId page.

viewJobExecHistory

Enter the name of a job to retrieve a history of job invocations. A table is displayed showing the job invocations (the last invocation is shown first).

For each invocation the number of triggered CIs and the total running time is shown. The Execution Details column shows at which times the job was executed. If the Probe shut down in the middle of a job execution and then resumed running or if there were blackout periods during the job execution, several running times are shown.

viewJobProblems

Enter the name of a job to retrieve a list of Trigger CIs that have problems for that job. Enter the name of a Trigger CI to retrieve a list of problems for that trigger CI. If no values are entered, problems all the jobs and triggers are displayed.

Column	Description
Job ID	Displayed if the **jobID** field is left empty.

Column	Description
	The job name as it appears in Data Flow Management. Click a job to go to its viewJobStatus page, to view its status and scheduling information.
Trigger CI	Displayed if the **triggerID** field is left empty. The CMDB object ID of the trigger for a job.
ErrMsgCode	The error message hash string (error hash ID).
ErrParams	The error parameters.
Severity	The severity of the error.

viewJobResultCiInstances

Fill in one or more of the parameters to return a list of CIs that have been discovered by a job.

The **Object State Holder** column displays the code for the CI or relationship defined in the CMDB. For details on the **appilog.common.system.typesClass ObjectStateHolder** method, see the **ObjectStateHolder** method in the online API documentation.

viewJobResults

Fill in one or more of the parameters to return a list of CIs that have been discovered by a job.

When **Hide Touched CIs Info** is set to **True**, the results page displays the following information:

Column	Description
Job Name	Displayed if the **jobID** field is left empty. The job name as it appears in Data Flow Management. Click a job to go to its viewJobStatus page, to view its status and scheduling information.
CI Type	Click to filter the list to show results for one CIT only.
Total CIs	Click to go to the viewJobResultCiInstances page, to view a list of all CIs that have been discovered by a job.
Triggered CIs	Click to go to the viewJobTriggeredCIs page, to view a list of all Trigger CIs that have been discovered by a job.
Last Discover Time	The date and time that the job was invoked.

When **Hide Touched CIs Info** is set to **False**, the results page displays the following information:

Column	Description
Job Name	Displayed if the **jobID** field is left empty. The job name as it appears in Data Flow Management. Click a job to go to its viewJobStatus page, to view its status and scheduling information.
CI Type	Click to filter the list to show results for one CIT only.
Touched CIs	Click to go to the viewJobResultCiInstances page, to view a list of those CIs discovered by the job that are **Touched CIs**.
Non Touched CIs	Click to go to the viewJobResultCiInstances page, to view a list of those CIs discovered by the job that are not Touched CIs.
Triggered CIs for Touched CIs	Click to go to the viewJobTriggeredCIs page, to view a list of those Trigger CIs included in a job that are Touched CIs.
Triggered CIs for Non Touched CIs	Click to go to the viewJobTriggeredCIs page, to view a list of those Trigger CIs included in the job that are not Touched CIs.
Last Discover Time	The date and time that the job was invoked.

You can further filter results in the results page by entering text filters in one of the fields, and clicking the **Search** button.

viewJobsStatuses

Click the **viewJobsStatuses** button to return status and scheduling information for all jobs. You can choose to filter the results.

> **Note:** This page is saved under **DataFlowProbe\runtime\jobsStatuses** once a day.

The results page displays the following information:

Column	Description
No.	The number of the job in the list.
Job Name	The job name as it appears in Data Flow Management. Click a job to go to its viewJobStatus page, to view its status and scheduling information.
Status	The severity of the job's status, as calculated by the Probe.

Column	Description
	• **Blocked.** Not in use. • **Removed.** The job is no longer active. • **Done/Total Triggers.** The number of trigger CIs that the Probe finished running on, against the total number of triggers for the job. For example, **(28/69)** indicates that there is a total of 69 triggers for the job, while the Probe has completed running on 28 of those triggers. • **Scheduled.** The job is scheduled to run. A red background signifies that a thread has run longer than expected and may be stuck. A green background signifies that the job is running as expected.
Triggered CIs	The Trigger CIs that have been run by the job. Click to go to the viewJobTriggeredCIs page.
Errors & Warnings	The number of errors and warnings for a specific job. Click to go to the viewJobErrorsSummary page, to view a list of error and warning messages reported on this job.
Last Invocation	The date and time that the job was last run.
Next Invocation	The date and time that the job is next scheduled to run.
Last Total run duration (seconds)	The length of time, in seconds, taken to run the job in the previous invocation. This is calculated according to the start time of the first trigger until the end time of the last trigger, even if triggers were added later on.
Avg run duration (seconds)	The average duration (in seconds) per trigger of the time it took the Probe to run this job.
Recurrence	The number of times that the job was invoked. Click to go to the viewJobExecHistory page, to retrieve a history of job invocations.
Results	The number of CITs that have been discovered by the job. Click to go to the viewJobResults page to view the CITs. **Note:** Displayed when **hideResults** parameter is set to **False**.

viewJobStatus

Enter the name of a job to return its status and scheduling information.

The results page displays the following information:

Column	Description
Threading info	The total number of worker threads created by the invocation, the free worker threads, and the stuck worker threads.
Total work time	The time that the Probe took to run this job.
Tasks waiting for execution	A list of jobs together with the number of Trigger CIs that are awaiting activation.
Max. Threads	The number of threads that are serving this job.
Progress	A summary of the current run, that is, since the specific run was activated. For example, `Progress: 2017 / 6851 destinations (29%)` means that out of 6851 CIs, 2017 CIs have already run.
Working Threads information	• **Thread Name.** The thread that is now running this job. Click to go to the **viewJobThreadDump** page. You use this page when a thread is running for a long time, and you must verify that this is because the thread is working hard, and not because there is a problem. • **Curr Dest. ID.** The name of the node on which the job is running. • **Curr Dest. IP.** The IP for which the job is discovering information. • **Work Time (Sec).** The length of time that this thread is running. • **Communication Log.** Click to go to the **viewCommunicationLog** page, to view an XML file that logs the connection between the Probe and a remote machine.
Discovery Jobs Information table	• **Status.** The severity of the job's status, as calculated by the Probe. For details, see "Status" on page 69. • **Triggered CIs.** Click to go to viewJobTriggeredCIs page, to view a list of Trigger CIs that are part of a job. • **Errors & Warnings.** Click to go to viewJobErrorsSummary page, to view a list of error and warning messages reported on this job. • **Last invocation.** The date and time that the job was last run. • **Next invocation.** The date and time that the job is next scheduled to run. • **Last Total run duration (seconds).** The length of time, in seconds, taken to run the job in the previous invocation. This is calculated according to the start time of the first trigger until the end time of the last trigger, even if triggers were added later on. • **Avg run duration (seconds).** The average duration (in seconds) per trigger of the time it took the Probe to run this job.

Column	Description
	• **Recurrence.** The number of times that the job was invoked. Click to go to viewJobExecHistory page, to view a history of job invocations.

Note: Click **Results** below the table to go to the viewJobResults page to view the CITs that have been discovered by the job.

viewJobTriggeredCIs

Fill in one or more of the parameters to return a list of Trigger CIs that are part of a job.

The results page displays the following information:

Note: Depending on the triggers, other information might also be displayed.

Column	Description
No.	The number of the job in the list.
Triggered CI ID	The CI instances that have been discovered by the job. Click to go to the viewJobTriggeredCIs page to view information about their CITs.
Last Execution Start Time	The date and time that the job last started running.
Last Execution End Time	The date and time that the job last finished running.
Service Exec. Duration (ms)	The maximum time that it took for a job to run in the last invocation, not including periods when the job did not run. Compare this result with the total execution duration. For example, when several jobs run simultaneously, but there is only one CPU, a job might have to wait for another job to finish. The service duration does not include this waiting time, whereas the total duration does.
Total Exec. Duration (ms)	The time that it took for a job to run in the last invocation, including the periods when the job did not run.
Last Run Status	The status of the last run, that is, whether the run succeeded or failed. In case of failure, click to go to the viewJobProblems page, to view a list of Trigger CIs with problems.

Column	Description
Priority	The priority of the job.
	Note: The lower the value, the higher the priority.

viewJobTriggeredCIsWithErrorId

Note: This operation is part of the inner interface and serves as a helper function. Do not use this page to view Trigger CIs information; instead, use the viewJobTriggeredCIs page.

The following is a list of parameters used in the above operations:

- **ciType.** The name of the CI type (for example, `ip`, `host`).

- **data.** A textual field in the **DiscoveryResults** table that contains information about the discovered object. For example:

```
<object class="ip">
<attribute name="ip_probename" type="String">EBRUTER02</attribute>
<attribute name="ip_address" type="String">16.59.58.200</attribute>
<attribute name="ip_domain" type="String">DefaultDomain</attribute>
</object>
```

- **Error Id.** The error message hash string (error hash ID) that is displayed in the **Jobs_Problems** table.

- **HideRemovedJobs.True**: do not display jobs that have run previously and are not relevant to the current run.

- **Hide Touched CIs Info.** Touched CIs are CIs which were discovered in previous invocations. DFM already has information about these CIs, so there is no need for the Probe to send the information to the server again. The server identifies that these CIs are relevant and that there is no need to enforce the aging mechanism on them. For details on aging, see "The Aging Mechanism Overview" in the *HPE Universal CMDB Administration Guide*.

  True: the table displays the total number of CIs and the total number of Trigger CIs for each CIT. **False**: The table displays the total number of CIs and Trigger CIs divided between touched CIs and non-touched CIs.

- **includeNonTouched.** Enables filtering the table to view non-touched CIs. Choose between viewing non-touched CIs only, all CIs (touched and non-touched), or none:

	Non-touched CIs		All CIs		No CIs	
(boolean)includeTouchedCis	○ True	⊙ False	⊙ True	○ False	○ True	⊙ False
(boolean)includeNonTouchedCis	⊙ True	○ False	⊙ True	○ False	○ True	⊙ False

- **includeNonTouchedCIs.** See **includeNonTouched**.

- **includeTouched.** Enables filtering the table to view touched CIs. Choose between viewing touched CIs only, all CIs (touched and non-touched), or none.

- **includeTouchedCIs.** See **includeTouched**.

- **jobID.** The name of the job, for example, **Host Applications by PowerShell**:
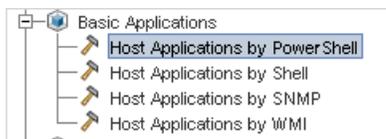
- **maxRows.** The maximum number of rows that should be displayed in the results table. The default is 100 or 1000.

- **maxTriggeredCIs.** See **maxRows**.

- **objectID.** The CMDB object ID.

- **hideRemovedJobs.** Hides information about jobs with the status, **REMOVED**. These are jobs that have run previously but that are not currently scheduled to run.

- **hideResults.** Indicates whether or not to hide the **Results** column. If the **Results** column is present, you can navigate to the job results.

- **triggerCI.** The CMDB object ID of the trigger for a job.

- **triggeredCiID.** See **triggerCI**.

How to View Discovery Rules

The Discovery Rules Engine is very large. You can search the rule base using search commands on the JMX console.

To search for a rule:

- Log in to the JMX console using the server administrator credentials

- Go to the service: **Normalization Rule Base Services**, and enter one of the following search commands:

Command	Description
scanForSNMPRules	Retrieves SNMP discovery rules that apply to the specified input attributes. **Note:** ○ the **sys_object_id** value must always have a leading "." ○ Leave empty to ignore
scanForScanFileRules	Retrieves Scan File discovery rules that apply to the specified input attributes. **Note:** Leave empty to ignore
viewNormalizationRuleById	Retrieves discovery rules by ID
viewNormalizationRuleByNiceId	Retrieves discovery rules by user friendly ID (NiceRuleID), **Example:** 4323@SNMP
viewNormalizationRules	Retrieves discovery rule outputs that apply to the specified input attributes **Format:** ○ Pair attributes in the following format: **attrName;attrValue** ○ Pairs must be separated by commas. **Example:** Name;HP,Version;10

How to View Discovery Resource History

Discovery resources are saved to the URM on the UCMDB Server and from there are distributed to all the Data Flow Probes.

Whenever a user changes the definition of a resource, an updated version of the resource is stored in the URM. The URM keeps all the historical revisions of each resource.

You can view changes between an older version and the current version of resources such as discovery scripts, integration and discovery adapters, discovery jobs, and so on, from the JMX console of the UCMDB Server.

Note: The purpose of this task is to describe how to access the discovery resources in the JMX

> console for the purpose of **viewing** the resources and their history only.
>
> **Adding or modifying a discovery resource in the JMX console is not supported.**

To view a discovery resource and its history:

1. Log in to the UCMDB JMX console.

2. In the UCMDB JMX Quick Search box, enter **listResourceTypes**.

3. Enter your Customer ID. (**Default: 1**)

4. Click **Invoke**. The **URM Services** mbean is displayed.

5. Among the UCMDB resource types displayed on this page, the following discovery resource types are displayed:

Resource Type	Description	Displays Diff Metadata	Displays Diff Content
Discovery_ADAPTER_ METADATA	Adapter resources	✓	✓
Discovery_ CONFIGURATION_FILE_ METADATA	Configuration Files	✓	✓
Discovery_JOB_METADATA	Discovery job definitions	✓	✓
Discovery_MODULE_ METADATA	Discovery modules	✓	✓
Discovery_WIZARD_ METADATA	Activity types	✓	✓
Discovery_SCRIPT_ METADATA	Script resources	✓	✓
Discovery_BIN_RESOURCE_ METADATA	External resources	✓	✗
Discovery_DOC_METADATA	PDF documents that come with the adapters	✓	✗
Discovery_MULTI_ SCANNER_METADATA	Multiple scanner packages	✓	✗
Discovery_SCANNER_ CONFIG_METADATA	Scanner configuration files	✓	✗
Discovery_SAI_RES_ METADATA	SAI resources	✓	✗

6. Click a resource type to view all the resources of that type.

7. To see the history of a particular resource, click the **history** link in that resource's row.

A page opens displaying the current version of the resource, as well as all its previous revisions.

8. Click the **Diff Content** link to view the actual change. All changes between the selected and current revisions are displayed.

> **Note:** The **Diff Content** link appears only for those resources whose changes you are able to see (see table above).

How to View Discovery Status of an Inventory CI in JMX

As an inventory discovery administrator, you can invoke the **getDiscoveryStateForInventoryCI** method to check the discovery status of an inventory CI, so that you can verify the health and live state of that device.

To do so,

1. Log in to the UCMDB JMX console. (Launch the Web browser and enter the following address: **https://<server_name>:8443/jmx-console**. You may have to log in with a user name and password.)

2. Locate **UCMDB:service=Discovery Manager** and click the link to open the Operations page.

3. Locate the **getDiscoveryStateForInventoryCI** operation.

4. Enter your Customer ID. (**Default: 1**)

5. For the **ciId** field, enter the global ID for the inventory CI for which you want to check the discovery status.

    a. In UCMDB UI, go to **Modeling > IT Universe Manager**.

    b. In the CI Selector pane, go to the **Search CIs** tab, click the **Start the Search** 🔍 icon for the **CI Name** field.

    c. From the returned results, select a discovered node.

    d. Go to the **Properties** tab in the right pane, scroll down to locate the **Global Id** property.

    e. Right-click the value for the **Global Id** property, select **Copy Property Name and Value**.

    f. Paste the value into the **cild** field in the JMX console and remove the property name.

6. Click **Invoke**.

   The result displays, which looks similar to the following:

   JMX Search   JMX List   Operations Index   Back to MBean   Reinvoke MBean   (Current Server is a writer: SGDLITVM0420)
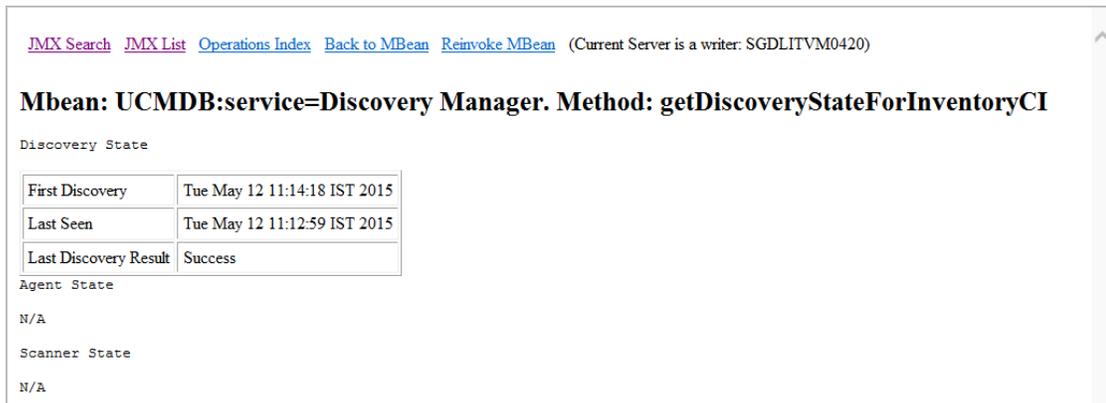
   Mbean: UCMDB:service=Discovery Manager. Method: getDiscoveryStateForInventoryCI

   Discovery State

First Discovery	Tue May 12 11:14:18 IST 2015
Last Seen	Tue May 12 11:12:59 IST 2015
Last Discovery Result	Success

   Agent State

   N/A

   Scanner State

   N/A

   The result page contains discovery information in three sections:

○ **Discovery State**

Field	Description	
First Discovery	Time when the CI was discovered for the first time.	
Last Seen	Time when this CI was last discovered successfully by a job. When there are no active jobs to discover this CI, or all jobs failed in discovering this CI, the value is **N/A**.	
Last Discovery Result	Last discovery result. Available statuses include: • **Pending.** There are one or more active discovery jobs that are running. This CI is not yet discovered. • **Success.** The CI was successfully discovered. • **Fail.** Discovery of this CI failed with one or more errors. • **Warning.** Discovery of this CI completed with one or more warning messages. • **N/A.** No active jobs and trigger CIs are available for discovering this CI.	

○ **Agent State**

N/A if no agent is available.

When agent is used, this section displays agent related information. The same information can also be found in the Properties tab of the CI.

○ **Scanner State**

N/A is no scanner is available.

When scanner is used, this section displays scanner related information. The same information can also be found in the Properties tab of the CI.

Note:

• If the trigger CI is a Probe Gateway CI, then the job discovery status and the **Last Seen** value for the specific CI will not display.

• When a discovery job is deactivated, its last discovery record is not available even if the job had successfully discovered the CI when it was active.

• The Last Seen value or the Last Discovery Result occasionally get reflected a bit late after the job is finished in the Universal Discovery UI.

How to View Agent Deployment Log for an Inventory CI in JMX

As an inventory discovery administrator, you can invoke the **getAgentDeploymentLogForInventoryCI** method to check the agent deployment log for an inventory CI. You work with the JMX console.

This task includes the following steps:

1. Log in to the UCMDB JMX console. (Launch the Web browser and enter the following address: **https://localhost:8443/jmx-console**. You may have to log in with a user name and password.)

2. Locate **UCMDB:service=Discovery Manager** and click the link to open the Operations page.

3. Locate the **getAgentDeploymentLogForInventoryCI** operation.

4. In the **Value** field for **customerID**, enter you customer ID. (**Default: 1**)

5. For the **ciId** field, enter the global ID for the inventory CI for which you want to check the discovery status.

   a. In UCMDB UI, go to **Modeling > IT Universe Manager**.

   b. In the CI Selector pane, go to the **Search CIs** tab, click the **Start the Search** [icon] icon for the **CI Name** field.

   c. From the returned results, select a discovered node.

   d. Go to the **Properties** tab in the right pane, scroll down to locate the **Global Id** property.

   e. Right-click the value for the **Global Id** property, select **Copy Property Name and Value**.

   f. Paste the value into the **ciId** field in the JMX console and remove the property name.

6. Click **Invoke**.

   The result displays.

How to View and Track Hotfixes Applied on Probe

Version 10.30 introduced the **showAllBinariesApplied** JMX method, allowing you to easily view and track all information about the infra hotfixes you deployed on the Probe.

To view and track hotfixes deployed on probe,

1. On the Data Flow Probe, launch the Web browser and enter the following address:
   http://localhost:1977

2. Enter the JMX console authentication credentials (default login name = **sysadmin**).

3. Locate the **showAllBinariesApplied** operation in the **GeneralUtils** category:

4. Click **Invoke**.

   The result page of the **showAllBinariesApplied** operation displays detailed information about the hotfixes applied, including hotfix type, readme file, if there're any overlapping files, and if there are any wrongly placed files.

How to Run Data Flow Ad Hoc Updates

1. Log in to the UCMDB JMX console. (Launch the Web browser and enter the following address: **https://localhost:8443/jmx-console**. You may have to log in with a user name and password.)

2. Click **UCMDB:service=Discovery Manager** to open the JMX MBEAN View page.

3. Run one of the following methods, depending on which is relevant:

JMX Method	Description
recalculateAndUpdateDFMTasks	Updates data flow tasks for all the adapters for which data flow task update is enabled. **Note:** Data flow task updates are enabled in the adapter's configuration file.
recalculateAndUpdateDFMTasksForAdapter	Updates data flow tasks for selected adapters without checking the adapter configurations. That is, even if the data flow task update is not enabled for a selected adapter, the updates are run.

How to Delete Unsent Probe Results

This task describes how to empty the Probe queue that contains results that have not yet been transmitted to the UCMDB Server.

1. Access the Data Flow Probe JMX console: Launch a Web browser and enter the following address: **http://<Probe Gateway machine name or IP address>:1977**. If you are running the Data Flow Probe locally, enter **http://localhost:1977**.

   You may have to log in with a user name and password.

2. Locate the **Probe_<Probe Name> > type=MainProbe** service and click the link to open the JMX MBEAN View page.

3. Invoke the operation by clicking the **dropUnsentResults** button.

> **Note:** This operation deletes 100 results at a time. To delete more results, re-invoke the operation as many times as necessary.

How to Configure Global ID Generation

1. Launch the Web browser and enter the following address:
   https://<CMS server>:8443/jmx-console.

2. Click **UCMDB:service=Multiple CMDB Instances Services** to open the JMX MBEAN View page.

3. Click one of the following methods and enter values as required:

setAsGlobalIdGenerator	Specifies that the CMDB will act as the global ID generator for all locally existing scopes
setAsGlobalIdGeneratorForScopes	Specifies the scopes for which global IDs will be generated
setAsNonGlobalIdGenerator	Stops the CMDB from acting as the global ID generator for all scopes

4. Click **Invoke**.

> **Note:** If you want to check which scopes are currently set, use the **getGlobalIdGeneratorScopes** method.

How to Perform Initial UCMDB-UCMDB Synchronization

This procedure performs a full synchronization of CIs and relations between CMDBs, while retaining the original CMDB IDs. CIs are replicated from the external CMS to the UCMDB. The procedure is generally intended to be performed only once, on a new system.

1. Launch a Web browser that connects to the CMS, and enter the following address: **https://<CMS server>:8443/jmx-console**.

2. Click **UCMDB:service=Multiple CMDB Instances Services** to open the JMX MBEAN View page.

3. Click the **fetchAllDataFromAnotherCMDB** method.

4. Enter values as required for the following fields:

   > **Note:** You must enter information in fields that do not have default values.

   - Customer ID

   - Remote user name

   - Remote password

   - Remote host name

   - Remote port **8443**

   - Remote Customer name (default value is **Default Client**)

   - Maximum chunk size

   - CI type to sync (default value is **managed_object**, causing all CI types to be synchronized)

   - Relation type to sync (the default value is **managed_relationship**, causing all relation types to be synchronized)

5. Click **Invoke**.

How to Export and Import Management Zones

To export and import a management zone using JMX console,

1. Log in to the UCMDB JMX console. (Launch the Web browser and enter the following address: **https://<server_name>:8443/jmx-console**. You may have to log in with a user name and password.)

2. Locate **UCMDB:service=Discovery Manager** and click the link to open the Operations page.

3. Locate the **getManagementZone** operation.

4. Enter your Customer ID. (**Default: 1**)

5. In the **Management Zone name** field, enter name of the management zone that you want to export.

6. Click **Invoke**.

   The result displays. In the Resource XML text box, press **Ctrl+A** and then **Ctrl+C** to select and copy all text.

7. Go back to the **UCMDB:service=Discovery Manager** Operations page, and locate the **addManagementZone** operation.

8. Enter your Customer ID. (**Default: 1**)

9. In the **Management Zone XML** value field, press **Ctrl+V** to paste the XML you just copied.

10. Click **Invoke**.

Data Flow Probe Log Files

Data Flow Probe logs store information about job activation that occurs on the Probe Gateway and Probe Manager. The log files can be accessed from the following location:

C:\hp\UCMDB\DataFlowProbe\runtime\log

> **Note:** Alternatively, to access the Data Flow Probe's log files, log in to the JMX console (http://<probe_machine>:1977/jmx-console/) and, from the main page, select the **GeneralUtils** mbean. Activating the **executeLogGrabber** function zips all the Data Flow Probe's log files. Save the .zip file locally on your client machine.

General Logs

WrapperProbeGw.log	Records all the Probe's console output in a single log file.
	• **Levels:**

	◦ **Error.** Any error that occurs within the Probe Gateway. ◦ **Information.** Important information messages, such as the arrival or removal of a new task. ◦ **Debug.** N/A • **Basic Troubleshooting:** Use this file for any Probe Gateway problems to verify what occurred with the Probe Gateway at any time as well as any important problems it encountered.
probe-error.log	Summary of the errors from the Probe. • **Levels:** ◦ **Error.** All errors in the Probe components. ◦ **Information.** N/A ◦ **Debug.** N/A • **Basic Troubleshooting:** Messages from the Probe's infrastructure only.
wrapperLocal.log	When running the Probe in separate mode (that is, the Probe Manager and Probe Gateway are installed on separate machines), a log file is also saved to the Probe Manager. • **Levels:** ◦ **Error.** Any error that occurs within the Probe Manager. ◦ **Information.** Important information messages such as received tasks, task activation, and the transferring of results. ◦ **Debug.** N/A • **Basic Troubleshooting:** Use this file for any Probe Manager problems to verify what occurred with the Probe Manager at any time as well as any important problems it encountered.
postgresql.log	Displays database related error during the installation. **Note:** If this log is empty check in the Event Viewer logs.

Probe Gateway Logs

probeGW-taskResults.log	Records all the task results sent from the Probe Gateway to the server. • **Levels:** ◦ **Error.** N/A ◦ **Information.** Result details: task ID, job ID, number of CIs to delete or update.

	○ **Debug.** The **ObjectStateHolderVector** results that are sent to the server (in an XML string).
	• **Basic Troubleshooting:**
	○ If there is a problem with the results that reach the server, check this log to see which results were sent to the server by the Probe Gateway.
	○ The results in this log are written only after they are sent to the server. Before that, the results can be viewed through the Probe JMX console (use the **ProbeGW Results Sender** MBean). You may have to log in to the JMX console with a user name and password.
probeGW-tasks.log	Records all the tasks received by the Probe Gateway.
	• **Levels:**
	○ **Error.** N/A
	○ **Information.** N/A
	○ **Debug.** The task's XML.
	• **Basic Troubleshooting:**
	○ If the Probe Gateway tasks are not synchronized with the server tasks, check this log to determine which tasks the Probe Gateway received.
	○ You can view the current task's state through the JMX console (use the **Discovery Scheduler** MBean).

Probe Manager Logs

probeMgr-performance.log	Performance statistics dump, collected every predefined period of time, which includes memory information and thread pool statuses.
	• **Levels:**
	○ **Error.** N/A
	○ **Information.** N/A
	○ **Debug.** N/A
	• **Basic Troubleshooting:**
	○ Check this log to investigate memory issues over time.
	○ The statistics are logged every 1 minute, by default.
probeMgr-adaptersDebug.log	Contains messages that are created following a job execution.

Discovery Rules Engine Log Files

normalization.audit.log	Logs information about the processing of the Discovery Rules Engine.
	- **Levels:**
	○ **Error.** N/A
	○ **Information.** Audits the number of element processed and the number of CIs that were changed.
	Example:
	Normalization (OSHV: 8 elements) (Time: 125 ms) (Modified CIs: 1)
	○ **Debug.** N/A
normalization.log	Logs detailed information about the processing of the Discovery Rules Engine, enabling you to trace detailed information of the Discovery Rule Engine process.
	- **Levels:**
	○ **Error.** All discovery rule processing errors.
	○ **Information.** Logs all levels of information about the processing of the Discovery Rules Engine.
	○ **Debug.** Logs mainly for debugging purposes.
	- **Basic Troubleshooting.** Check this log when you need to analyze why a CI was not enriched by the Discovery Rules Engine.

How to Check XML Enricher Health Using JMX

This task describes how to view health statistics of an XML Enricher service using the JMX console.

1. Prerequisites

   The Data Flow Probe where the XML Enricher service is running is started.

2. Connect to the Data Flow Probe

   Launch your Web browser and enter the following address: **http://<*DataFlowProbe*>:1977**, where **<DataFlowProbe>** is the name of IP address of the machine on which the XML Enricher service is running.

3. View statistics

   a. Under the **Local_<DataFlowProbe>** section, click the **XMLEnricherMonitor** service.

   b. Select the **viewXmlEnricherStatuses method** and click **Invoke**.

4. Results

   Health statistics for the XML Enricher are displayed.

How to Check the Confidential Manager Connection

1. Access the Data Flow Probe JMX console by launching a Web browser and entering the following address: **http://<Probe Gateway machine name or IP address>:1977**. If you are running the Data Flow Probe locally, enter **http://localhost:1977**.

   You may have to log in with a user name and password.

2. Enter **isCMClientInitialized** in the quick search field and click the link that appears.

3. Click **Invoke**.

   The current status of the Confidential Manager server is displayed.

How to Increase the Number of Threads for Data Push Jobs

Version 10.30 introduced a new JMX setting **com.hp.ucmdb.synchronizer.manager.SynchronizerManagerFactory**, allowing you to increase the number of threads for data push jobs from the out-of-the-box value **3** to any desired value.

How to increase the number of threads for data push jobs

> **Note:** Apply this change for large UCMDB deployments where the hardware resources for UCMDB server and data flow probe are assigned as documented in the *HPE Universal CMDB Sizing Guide*.

To do so,

1. Log in to the UCMDB JMX console. (Launch the Web browser and enter the following address: **https://<server_name>:8443/jmx-console**. You may have to log in with a user name and password.)

2. Enter **setInternalSetting** in the Quick Search field and click the link that appears with the **UCMDB:service=Settings Services** category.

3. Invoke the **setInternalSetting** JMX method with the following parameters:

   ○ **customerID:** Enter your Customer ID. (**Default: 1**)

   ○ **key:** Enter **com.hp.ucmdb.synchronizer.manager.SynchronizerManagerFactory**.

   ○ **value:** Enter a desired value as the number of threads for push jobs. (**Default: 3**).

4. Double check the value set by invoking the **getInternalSettings** JMX method.

   The returned result contains the **com.hp.ucmdb.synchronizer.manager.SynchronizerManagerFactory** setting and its value.

5. Restart the UCMDB Server.

6. The new value is loaded successfully if you see the following message logged in the **startup.log** file:

```
The number of threads for manager
com.hp.ucmdb.synchronizer.manager.SynchronizerManagerFactory was overridden
to ...
```

   Note: The **startup.log** file is located in the **<UCMDB_Install_dir>\runtime\logs** directory.

Revert the change

To revert the change and use the default value,

1. Perform either of the following from the UCMDB JMX console:

   ○ Invoke the **setInternalSetting** JMX method with the following parameters:

     • **customerID:** Enter your Customer ID. (**Default: 1**)

     • **key:** Enter **com.hp.ucmdb.synchronizer.manager.SynchronizerManagerFactory**.

     • **value:** Enter the default value **3**.

   ○ Delete the new setting by invoking the **listResources** method from the **UCMDB:service=URM Services** category.

     If you delete the new setting, UCMDB will use the hard-coded default value.

     To do so,

i. Enter **listResources** in the Quick Search field and click the link that appears with the **UCMDB:service=URM Services** category.

ii. Invoke the **listResources** method with the following parameters:

- **customerID:** Enter your Customer ID. (**Default: 1**)

- **resourceType:** Enter **Settings_STATE_CUSTOMER_SETTING**.

iii. On the returned result page, click **delete** for the **com.hp.ucmdb.synchronizer.manager.SynchronizerManagerFactory** setting.

2. Restart the UCMDB Server for the change to take effective.

How to Set a Default List of CyberArk Properties Using JMX

The enhanced CyberArk integration in version 10.30 allows you to configure selected CyberArk properties as a query string for UCMDB/UD to retrieve passwords from the CyberArk Enterprise Password Vault. The out-of-the-box list of CyberArk properties displayed in the Configure dialog box is editable, you can set selected CyberArk properties as the default list to display by using the new parameter **cyberark.parameter.list** of the **setGlobalSettingVaule** JMX method.

This task describes how to set a default list of CyberArk properties for the CyberArk integration using JMX console.

1. Access the UCMDB JMX console

   Launch a Web browser and enter the following address: **https://<UCMDB machine name or IP address>:8443/jmx-console**. You may have to log in with a user name and password.

2. (Optional) Display the OOTB list of CyberArk properties

   a. Search and locate the **getSettingDefaultValue** operation in the **UCMDB:service=Settings Services** category.

   b. Invoke the **getSettingDefaultValue** operation using the following parameter values:

   - **customerId:** 1

   - **name:** cyberark.parameter.list

3. Set the default list of CyberArk properties

   a. Search and locate the **setGlobalSettingValue** operation in the **UCMDB:service=Settings Services** category.

b. Invoke the **setGlobalSettingValue** operation using the following parameter values:

- **name:** cyberark.parameter.list

- **value:** Specify a list of CyberArk properties, separated by comma, for example, **Folder,Port,Domain**.

setGlobalSettingValue

Set global setting value

Name	Type	Value	Description
name	java.lang.String	cyberark.parameter.list	Setting Name
value	java.lang.String	Folder,Port,Domain　✕	Setting Value

Invoke

The change takes effective right away, and the OOTB list of CyberArk properties is overridden with this new list.

4. Check the CyberArk properties list in the Configure dialog box

If the Configure dialog box is already open, simply close it and open it again by clicking **Configure** Configure in the Protocol Parameters dialog box. Otherwise, follow the instructions below.

a. In UCMDB UI, go to **Data Flow Management > Data Flow Probe Setup > Domains and Probes > DefaultDomain(Default) > Credentials**.

b. Select a CyberArk integration enabled protocol, and open the Protocol Parameters dialog box.

c. Select **External Vault**, and then click **Configure** Configure .

The Configure dialog box now lists three CyberArk properties only.

○ Reference

⊙ Parameter   ☐ Regex

Name	Value
Folder	
Port	
Domain	
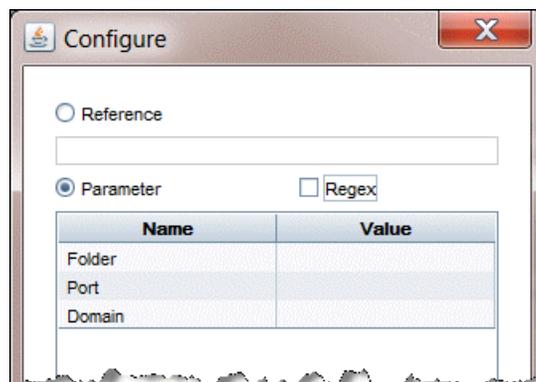
How to Enable Attribute Name Verification during the Matching Phase of Identification

The **reconciliation.match.attributes** JMX setting verifies names of attributes during the matching phase of the identification process. If, compared to the other CI, there is a different value among the CI attribute values defined in this setting, the verification process stops and the match is rejected.

Only the attributes that are inherited from the Managed Object should be used in this setting (for example, **global_id** and **name**). Names of the attributes specified in the setting should be separated by comma.

This setting applies only to CIs that are identified based on the Identification Rule.

The following example explains how this setting works:

- If you have two node CIs:

  - Node1 with **global_id**=g1 and **name**=n

  - Node2 with **global_id**=g2 and **name**=n

  - **reconciliation.match.attributes** set to **global_id**

  This scenario will result in the two nodes not matching.

- If Node1 does not have **global_id** set, the two CIs will match.

To enable this setting, do the following:

1. Go to **JMX Console > UCMDB:service=Settings Services > setSettingValue**.

2. In the **name** field, enter **reconciliation.match.attributes**.

3. In the **value** field, enter names of the CI attributes, separated by comma.

   > **Note:** To disable this setting, leave the **value** field empty.

4. Click **Invoke**.

How to Enable CI Type Tenant Owner Verification during the Matching Phase of Identification

You can dynamically add a verification criterion based on the **TenantOwner** attribute during the

matching phase of the identification process for CIs by using the new JMX setting **reconciliation.tenantaware.citypes**. If, compared to the other CI, there is a different value among the CI attribute values defined in this setting, the verification process stops and the match is rejected.

The following example explains how this setting works:

- If you have two node CIs:

  - Node1 with **TenantOwner**=t1 and **name**=n

  - Node2 with **TenantOwner**=t2 and **name**=n

  - **reconciliation.tenantaware.citypes** set to **node**

  This scenario will result in the two nodes not matching.

- If Node1 does not have **TenantOwner** set, the default Tenant will be provided.

> **Note:** If you add a node CIT, UNIX for example, you do not have to add it since it is inherited by identification rule. Only if you change the UNIX CIT identification rule, you need to add it explicitly.

To enable TenantOwner attribute verification for CITs that are identified by key attributes,

1. Add the **ID_ATTRIBUTE** qualifier for the **TenantOwner** attribute of the **managed_object** CIT.

   > **Note:** The **TenantOwner** attribute will be inherited to all the child classes. Make sure that a child class does not override it.

   a. Go to the **JMX console > UCMDB:service=URM Services**.

   b. Invoke the **listResourceTypes** method.

   c. On the returned page, click **CM_CLASS**, then click **managed_object**.

   d. In the Resource XML box, add the following attribute qualifier to the **TenantOwner** attribute:

   ```
   <Attribute-Qualifier name="ID_ATTRIBUTE" is-factory="true" is-user-
   modified="true" version="15" >
           <Data-Items/>
   </Attribute-Qualifier>
   ```

   e. Click **Save resource**.

2. Invoke the JMX method **updateClassModel** to update the class model (go to the **JMX console > UCMDB:service=Class Model Services**, and invoke the **updateClassModel** method with **1** in the **Value** field for customerID).

3. Reload the class model from persistency (go to the **JMX console > UCMDB:service=Class Model Services**, and invoke the **reloadClassModelFromPersistency** method).

4. Go to **JMX console > UCMDB:service=Model Services**, invoke the **recalculateID** method with **classname** field empty.

   This may take a while as it updates the calculated IDs for all instances of the classes with key attributes identification.

To enable the reconciliation.tenantaware.citypes setting for specific CITs (separated by comma) that are identified based on identification rules,

1. Go to **JMX Console > UCMDB:service=Settings Services > setSettingValue**.

2. In the **name** field, enter **reconciliation.tenantaware.citypes**.

3. In the **value** field, enter names of the CITs, separated by comma.

   | **Note:** To disable this setting, leave the **value** field empty.

4. Click **Invoke**.

To enable the reconciliation.tenantaware.citypes setting for all the CITs that are identified based on identification rules,

1. Go to **JMX Console > UCMDB:service=Settings Services > setSettingValue**.

2. In the **name** field, enter **reconciliation.tenantaware.citypes**.

3. In the **value** field, enter *****.

4. Click **Invoke**.

Note:

- If you changed the Tenant ID on the Data Flow Probe, make sure you clear the probe cache as well by performing either of the following:

  ○ Log in to the probe server, run the following script:

     Windows:\hp\UCMDB\DataFlowProbe\tools\clearProbeData.bat

     Linux:\hp\UCMDB\DataFlowProbe\tools\clearProbeData.sh

  ○ Log in to the UCMDB server UI, go to **Data Flow Management > Universal Discovery >**

> **Discovery Modules/Jobs**. For each of the jobs that run on the probe, right-click the job and select **Clear Probe Results Cache**.
>
> - In a multi-tenant aware environment, a tenant must be specified for the Data Flow Probe.

Tenant Owner Related Known Issues, Problems, and Workaround

- **PROBLEM:** After switching to Tenant aware reconciliation, the **OwnerTenant** attribute becomes read-only in the Configuration Item Properties dialog.

  Workaround: Use **Assign Tenants** functionality from the CI's context menu.

- **PROBLEM:** After removing the Key Attributes qualifier from the **OwnerTenant** attribute of the Managed Object, sometime no properties are displayed for the CIs in UI.

  Workaround: If you want to switch back (to disable Tenant aware reconciliation), do the following:

  a. Remove the **ID_ATTRIBUTE** qualifier for the **TenantOwner** attribute on the **managed_object** CIT.

  b. Remove the value of the **reconciliation.tenantaware.citypes** setting.

  c. Reload the class model from persistency (go to the **JMX console > UCMDB:service=Class Model Services**, and invoke the **reloadClassModelFromPersistency** method).

  d. Go to **JMX console > UCMDB:service=Model Services**, invoke the **recalculateID** method with **classname** field empty.

  e. Go to **JMX console > UCMDB:service=Model Services**, invoke the **updateClasModel** method.

- **LIMITATION:** Enrichment is not invoking the Reconciliation on Update **OwnerTenant** via **Associate Tenant Rule**. As a result, you may have duplicated data in the system in case if you update the **OwnerTenant**'s CI to a tenant that already has this CI.

  Workaround: None.

- **LIMITATION:** CIs with Identification rule would be duplicated in case if the user is updating the **OwnerTenant** CI to a tenant that already has this CI from **Update OwnerTenant** in the **Assign Tenants** module.

  Workaround: None.

- **PROBLEM:** When adding Consumer Tenants to a CI, the System Default Tenant appears in the

list of Consumers after saving, even if it was not selected. This issue occurs only when changing the Owner Tenant or the Consumer Tenant.

Workaround: None.

- **PROBLEM:** When removing all Consumer Tenants from a CI (from the IT Universe), an error is thrown and the Owner Tenant is overwritten with the System Default Tenant.

  Workaround: To avoid removing the System Default Tenant from the Consumer Tenants list, make sure you set the System Default Tenant as consumer.

  Only when the System Default Tenant is not set as consumer, the Owner Tenant will be overwritten with the System Default Tenant when trying to save.

- **PROBLEM:** Error message received when setting up a tenant aware environment, for the OOTB enrichments which are adding CIs. (QCCR1H104949)

  Workaround: If there are enrichments which are creating new CIs, after setting the environment as tenant aware, the attribute **Owner tenant** should be set for those CI Types which are being created through enrichments.

Chapter 5: Developer Reference Methods

This chapter includes:

How to Debug Adapter Resources

This task describes how to use the JMX console to create, view, and delete adapter state resources (any resources created using the resource manipulation methods in the DataAdapterEnvironment interface, which are saved in the UCMDB database or the Probe database) for debugging and development purposes.

1. Launch the Web browser and enter the server address, as follows:

   ○ For the UCMDB server: **https://localhost:8443/jmx-console**

   ○ For the Probe: **http://localhost:1977**

   You may have to log in with a user name and password (default user name: sysadmin).

2. To open the JMX MBEAN View page, do one of the following:

   ○ On the UCMDB server: click
   UCMDB:service=FCMDB Adapter State Resource Services

   ○ On the Probe: click **type=AdapterStateResources**

3. Enter values in the operations that you want to use, and click **Invoke**.

How to Create an Integration User

You can create a dedicated user for integrations between other products and UCMDB. This user enables a product that uses the UCMDB client SDK to be authenticated in the server SDK and execute the APIs. Applications written with this API set must log on with integration user credentials.

> **Caution:** It is also possible to connect with a regular UCMDB user (for instance, admin).

However, this option is not recommended. To connect with a UCMDB user, you must grant the user API authentication permission.

To create an integration user:

1. Launch the Web browser and enter the server address, as follows:

   `https://localhost:8443/jmx-console`

   You may have to log in with a user name and password.

2. Under `UCMDB`, click **service=UCMDB Authorization Services**.

3. Locate the **createUser** operation. This method accepts the following parameters:

   ○ **customerId**. The customer ID.

   ○ **username**. The integration user's name.

   ○ **userDisplayName**. The integration user's display name.

   ○ **userLoginName**. The integration user's login name.

   ○ **password**. The integration user's password.

     The default password policy requires the password to include at least one of each of the four following types of characters:

     • Uppercase alphabetic characters

     • Lowercase alphabetic characters

     • Numeric characters

     • Symbol characters **,\:/. _?&%=+-[]()|**

     It also requires the password to adhere to the minimum length, which is set by the **Password minimum length** infrastructure setting.

4. Click **Invoke**.

5. In a single-tenant environment, locate the **setRolesForUser** method and enter the following parameters:

   ○ **userName**. The integration user's name.

   ○ **roles.** SuperAdmin.

   Click **Invoke**.

6. Locate the **setUserServerAdministratorValue** method and enter the following parameters:

- ○ **customerID**. The customer ID.

- ○ **userLoginName**. The integration user's login name.

- ○ **serverAdministratorValue**. Select **True**.

Click **Invoke**.

7. In a multi-tenant environment, locate the **grantRolesToUserForAllTenants** method and enter the following parameters to assign the role in connection with all tenants:

- ○ **userName**. The integration user's name.

- ○ **roles.** SuperAdmin.

Click **Invoke**.

Alternatively, to assign the role in connection with specific tenants, invoke the **grantRolesToUserForTenants** method, using the same userName and roles parameter values. For the **tenantNames** parameter, enter the required tenants.

8. Either create more users, or close the JMX console.

9. Log on to UCMDB as an administrator.

10. From the **Administration** tab, run **Package Manager**.

11. Click the **Create custom package** icon.

12. Enter a name for the new package, and click **Next**.

13. In the Resource Selection tab, under **Settings**, click **Users**.

14. Select a user or users that you created using the JMX console.

15. Click **Next** and then **Finish**. Your new package appears in the Package Name list in Package Manager.

16. Deploy the package to the users who will run the API applications.

For details, see the section "How to Deploy a Package" in the *HPE Universal CMDB Administration Guide*.

> **Note:** The integration user is per customer. To create a stronger integration user for cross-customer usage, use a **systemUser** with the **isSuperIntegrationUser** flag set to **true**. Use the **systemUser** methods (**removeUser**, **resetPassword**, **UserAuthenticate**, and so on).
>
> There are two out-of-the-box system users. It is recommended to change their passwords after installation using the **resetPassword** method.

- ○ **sysadmin/sysadmin**

- ○ **UISysadmin/UISysadmin** (this user is also the **SuperIntegrationUser**).

  If you change the UISysadmin password using **resetPassword**, you must do the following:

  i. In the JMX Console, locate the **UCMDB-UI:name=UCMDB Integration** service.

  ii. Run **setCMDBSuperIntegrationUser** with the user name and new password of the integration user.

Web Service API - executeTopologyQueryWithParameters

The `executeTopologyQueryWithParameters` method retrieves a `topologyMap` element that matches the parameterized query.

The query is passed in the `queryXML` argument. The values for the query parameters are passed in the `parameterizedNodeList` argument. The TQL must have unique labels defined for each `CINode` and each `relationNode`.

The `executeTopologyQueryWithParameters` method is used to pass ad hoc queries, rather than accessing a query defined in the CMDB. You can use this method when you do not have access to the UCMDB user interface to define a query, or when you do not want to save the query to the database.

To use an exported TQL as the input of this method, do the following:

1. Launch the Web browser and enter the following address:
   http://localhost:8443/jmx-console.

   You may have to log in with a user name and password.

2. Click **UCMDB:service=TQL Services**.

3. Locate the **exportTql** operation.

   - ○ In the **customerId** parameter box, enter **1** (the default).

   - ○ In the **patternName** parameter box, enter a valid TQL name.

4. Click **Invoke**.

Chapter 6: Configuration Manager Methods

This chapter includes:

Configuration Manager JMX Methods

How to Enable Advanced Content

If you purchased the relevant license after deploying Configuration Manager, perform the following procedure to activate the content:

1. Launch your Web browser and enter the following address: **http://<server_name>:<port_ number>/cnc/jmx-console**, where **<server_name>** is the name of the machine on which Configuration Manager is installed.

2. Enter the JMX console authentication credentials.

3. Under **Configuration Manager**, click **ImportExport service**.

4. Locate the **activateAutomanageResource** operation and click **Invoke**.

How to Delete Advanced Content

If you want to delete advanced content that was previously installed, do the following:

1. Launch your Web browser and enter the following address: **http://<server_name>:<port_ number>/cnc/jmx-console**, where **<server_name>** is the name of the machine on which Configuration Manager is installed.

2. Enter the JMX console authentication credentials.

3. Under **Configuration Manager**, click **Licensed content service**.

4. Locate the **deactivateAutomanagedResources** operation and click **Invoke**.

How to Export the System Data

This task describes how to list and export the system data, views, and policies of Configuration Manager and store this information in its file system.

1. Launch your Web browser and enter the following address: **http://<server_name>:<port_ number>/cnc/jmx-console,** where **<server_name>** is the name of the machine on which Configuration Manager is installed.

2. Enter the JMX console authentication credentials.

3. Under **Configuration Manager**, click **ImportExport service**.

4. Locate one of the following operations:

   ○ **exportData**

   ○ **listAllViews**

   ○ **exportViews**

   ○ **listAllPolicies**

   ○ **exportPolicies**

5. In the **Value** field, enter the file name and the full path of the directory in the file system of the Configuration Manager server to which the data is exported. You can also provide a network path if you do not want the exported file to reside on the same server.

6. Click **Invoke** to export the data. The data is exported as an XML file to the specified directory.

How to Import the System Data

This task describes how to import the XML file containing the system data from Configuration Manager's file system to another Configuration Manager of the same version using the JMX console.

1. Launch your Web browser and enter the following address: **http://<server_name>:<port_ number>/cnc/jmx-console,** where **<server_name>** is the name of the machine on which Configuration Manager is installed.

2. Enter the JMX console authentication credentials.

3. Under **Configuration Manager**, click **ImportExport service**.

4. Locate the **importData** operation.

5. In the **Value** field, enter the file name and the full path of the directory in the file system of the

Configuration Manager server from which the data is imported. You can provide a network path to import data from a file which does not reside on the same server.

6. Click **Invoke** to import the data.

How to Update the Configuration Manager Folding Rules

After defining folding rules for composite CIs in UCMDB, execute the following JMX commands to update the folding rules in Configuration Manager:

1. Access the JMX console by launching your Web browser and entering the following address: **http://<server_name>:<port_number>/cnc/jmx-console,** where **<server_name>** is the name of the machine on which Configuration Manager is installed.

2. Enter the JMX console authentication credentials.

3. Click **Configuration Manager > View Service**. Select **updateFoldingRules** and click **Invoke**.

How to Enable Large Capacity

Configuration Manager supports working with up to 20,000 composite CIs in a single managed view. To enable this functionality, do the following:

> **Note:**
>
> - If you want to enable this functionality, it is recommended to install Configuration Manager on a server that has a minimum of 8 GB of memory (RAM).
>
> - Managed views that are based on dynamic TQL queries and result in more than 20,000 composite CIs are not supported.

1. To access the JMX console, launch your Web browser and enter the following address: **http://<server_name>:<port_number>/cnc/jmx-console,** where **<server_name>** is the name of the machine on which Configuration Manager is installed.

2. Enter the JMX console authentication credentials.

3. Click **Configuration Manager > View Service**. Select **supportLargeViews** and click **Invoke**.

4. In UCMDB, change the value of the TQL Group View Result Size setting to 500,000 (**Administration > Infrastructure Settings Manager > TQL Settings**).

5. Do one of the following:

○ If you use the HPE Universal CMDB Configuration Manager Windows service to start Configuration Manager, navigate to the **<Configuration_Manager_installation_ directory>/bin/** folder and double-click the **edit-server-0.bat** file. In the Java tab, increase the value of the Maximum memory pool parameter to 4096 or greater.

○ If you use the **start-server-0.bat** file to start Configuration Manager, edit the **start-server-0.bat** file and raise the value of the –Xmx parameter to 4096m or greater.

Configuration Manager Best Practices

This section provides Configuration Manager related best practices.

Increase TQL Fuses/Capacity When Installing Configuration Manager on an Environment with a Huge Number of CIs

When Configuration Manager is installed on an environment with a huge number of CIs, it might be possible that UCMDB is not accessible, even though UCMDB server is up and running and the status is "green", and that login is also possible but after that the java loads forever. This issue is caused by the fact that the TQL getting the license capacity received a large number of CIs and those needed to be chunked.

You can resolve the issue by increasing the TQL parameters related fuses/capacity via JMX,

1. Access the UCMDB JMX console: Launch a Web browser and enter the following address: **https://<UCMDB machine name or IP address>:8443/jmx-console**. You may have to log in with a user name and password.

2. Locate **UCMDB:service=Settings Services** and click the link to jump to the Operations table.

3. Locate the **setSettingValue** operation.

4. To increase TQL parameters related fuses/capacity, invoke the **setSettingValue** method with the following parameters:

   ○ **tql.max.result.size.layout.retrieval**

      • **customerID**: 1

      • **name**: tql.max.result.size.layout.retrieval

      • **value**: 300000 (current value is 200000)

   ○ **tql.resultutils.chunk.maxresultsize**

- **customerID**: 1

- **name**: tql.resultutils.chunk.maxresultsize

- **value**: 300000 (current value is 100000)

5. Restart the server.

Chapter 7: Hardening Methods

This chapter includes:

How to Change the System User Name or Password through the JMX Console

The JMX console uses system users, that is, cross-customer users in a multi-customer environment. You can log in to the JMX console with any system user name.

You change the password either through the JMX console or through the Server Management tool.

To change the default system user name or password through the JMX console:

1. Launch a Web browser and enter the following address: **https://localhost:8443/jmx-console**.

2. Enter the JMX console authentication credentials.

3. Locate **UCMDB:service=Authorization Services** and click the link to open the Operations page.

4. Locate the **resetPassword** operation.

   - In the **userName** field, enter **sysadmin**.

   - In the **password** field, enter a new password.

     Note: The default password policy requires the password to include at least one of each of the four following types of characters:

- Uppercase alphabetic characters

- Lowercase alphabetic characters

- Numeric characters

- Symbol characters ,\:/. _?&%=+-[]()|

It also requires the password to adhere to the minimum length, which is set by the **Password minimum length** infrastructure setting.

5. Click **Invoke** to save the change.

To change the default system user name or password through the Server Management tool:

1. **For Windows:** run the following file: **C:\hp\UCMDB\UCMDBServer\tools\server_ management.bat.**

   For Linux: Run **server_management.sh** located in the following folder: **/opt/hp/UCMDB/UCMDBServer/tools/**.

2. Log in to the tool with the authentication credentials: **sysadmin/sysadmin**.

3. Click the Users link.

4. Select the system user and click **Change password for logged-on user**.

5. Enter the old and new passwords and click **OK**.

How to Enable Mutual Certificate Authentication for SDK

This mode uses SSL and enables both server authentication by the UCMDB and client authentication by the UCMDB-API client. Both the server and the UCMDB-API client send their certificates to the other entity for authentication.

Note:

- The following method of enabling SSL on the SDK with mutual authentication is the most secure of the methods and is therefore the recommended communication mode.

- The keystore used for client SDK must be in Java Keystore (JKS) format. The Java Cryptography Extension KeyStore (JCEKS) or other formats are not supported.

- The keystore used for SDK must contain only one key-pair and nothing else in it. The password for this key-pair must be the same as the one for keystore.

1. Harden the UCMDB-API client connector in UCMDB:

    a. Access the UCMDB JMX console: Launch a Web browser and enter the following address: **https://<UCMDB machine name or IP address>:8443/jmx-console**. You may have to log in with a user name and password.

    b. Locate **UCMDB:service=Ports Management Services** and click the link to open the Operations page.

    c. Locate the **PortsDetails** operation and click **Invoke**. Make a note of the HTTPS with client authentication port number. The default is 8444 and it should be enabled.

    d. Return to the Operations page.

    e. To map the ucmdb-api connector to the mutual authentication mode, invoke the **mapComponentToConnectors** method with the following parameters:

        - **componentName**: ucmdb-api

        - **isHTTPSWithClientAuth**: true

        - All other flags: false

        The following message is displayed:

        Operation succeeded. Component ucmdb-api is now mapped to: HTTPS_CLIENT_ AUTH ports.

    f. Return to the Operations page.

2. Repeat step 1 for the **ping** component.

3. Make sure the JRE that runs the UCMDB-API client has a keystore containing a client certificate.

    > **Note:** The UCMDB-API client certificate must have the minimum size key no less than 2048 bits.

4. Export the UCMDB-API client certificate from its keystore.

5. Import the exported UCMDB-API client certificate to the UCMDB Server Truststore.

    a. On the UCMDB machine, copy the created UCMDB-API client certificate file to the following directory on UCMDB:

        C:\HP\UCMDB\UCMDBServer\conf\security

    b. Run the following command:

        ```
        C:\HP\UCMDB\UCMDBServer\bin\jre\bin\keytool.exe -import -v -keystore
        C:\HP\UCMDB\UCMDBServer\conf\security\server.truststore -file <exported
        ```

```
UCMDB-api client certificate> -alias ucmdb-api
```

   c. Enter the UCMDB Server Truststore password (default **hppass**).

   d. When asked, **Trust this certificate?**, press **y** and then **Enter**.

   e. Make sure the output **Certificate** was added to the keystore.

6. Export the UCMDB server certificate from the server keystore.

   a. On the UCMDB machine, run the following command:

```
C:\HP\UCMDB\UCMDBServer\bin\jre\bin\keytool.exe -export -alias hpcert
-keystore
C:\HP\UCMDB\UCMDBServer\conf\security\server.keystore
-file C:\HP\UCMDB\conf\security\server.cert
```

   b. Enter the UCMDB Server Truststore password (default **hppass**).

   c. Verify that the certificate is created in the following directory:

      C:\HP\UCMDB\UCMDBServer\conf\security\server.cert

7. Import the exported UCMDB certificate to the JRE of the UCMDB-API client truststore.

8. The certificate used by the API Client must contain in it's Common Name (CN) field the name of a user that's present in UCMDB.

   This user MUST have an EMPTY password and all required permissions for SDK access.

   To set an empty password to an existing UCMDB user,

   a. Go to **JMX Console > UCMDB:service=URM Services > listResourceTypes**.

   b. Click **Auth_USER**.

   c. Click your user and wait for the XML to load.

   d. In the XML, replace the password with **s39t3O*tfoZXg30xd/nvJGL5is8=**.

   e. Click **Save resource**.

9. Restart the UCMDB Server and the UCMDB-API client.

10. To connect from the UCMDB-API client to UCMDB-API server, use the following code:

```
UcmdbServiceProvider provider = UcmdbServiceFactory.getServiceProvider
("https", <SOME_HOST_NAME>, <HTTPS_WITH_CLIENT_AUTH_PORT_NUMBER
(default:8444>));
UcmdbService ucmdbService = provider.connect
(provider.createCertificateCredentials(<TheClientKeystore.
```

```
e.g: "c:\\client.keystore">, <KeystorePassword>),
provider.createClientContext(<ClientIdentification>));
```

How to Configure a Reverse Proxy

You can make changes to the reverse proxy configuration by using the JMX console on the HP Universal CMDB server machine. This configuration is only necessary when creating a direct link to a report using the Scheduler.

To change a reverse proxy configuration:

1. On the HP Universal CMDB server machine, launch the Web browser and enter the following address:

   https://localhost.<domain_name>:8443/jmx-console

   You may have to log in with the user name and password.

2. Click the **UCMDB-UI > UCMDB-UI:name=UI Server frontend settings** link.

   In the **setUseFrontendURLBySettings** field, enter the server proxy URL, for example, **https://my_proxy_server:443/**.

3. Click **Invoke**.

4. To see the value of this setting, use the **showFrontendURLInSettings** method.

How to Change the Server Keystore Password

After installing the Server, the HTTPS port is open and the store is secured with a weak password (the default **hppass**). If you intend to work with SSL only, you must change the password.

The following procedure explains how to change the **server.keystore** password only. However, you should perform the same procedure for changing the **server.truststore** password.

> **Note:** You must perform every step in this procedure.

1. Start the UCMDB Server.

2. Execute the password change in the JMX console:

a. Launch the Web browser on the UCMDB Server machine, as follows:
https://localhost:8443/jmx-console.

> **Note:** Starting from version 10.30, access to the JMX console is restricted to localhost only. If you need to access the JMX console remotely, see "How to Enable Remote Access to the JMX Console".

> **Note:** Starting from version 10.30, access to the JMX console is restricted to localhost only. If you need to access the JMX console remotely, see "How to Enable Remote Access to the JMX Console".

You may have to log in with a user name and password.

b. Under UCMDB, click **UCMDB:service=Security Services** to open the Operations page.

c. Locate and execute the **changeKeystorePassword** operation.

This field must not be empty and must be at least six characters long. The password is changed in the database only.

3. Stop the UCMDB Server.

4. Run commands.

From **C:\hp\UCMDB\UCMBServer\bin\jre\bin**, run the following commands:

a. Change the store password:

keytool -storepasswd -new <new_keystore_pass> -keystore C:\hp\UCMDB\UCMDBServer\conf\security\server.keystore -storepass <current_keystore_pass>

b. The following command displays the inner key of the keystore. The first parameter is the alias. Save this parameter for the next command:

keytool -list -keystore C:\hp\UCMDB\UCMDBServer\conf\security\server.keystore

c. Change the key password (if the store is not empty):

keytool -keypasswd -alias <alias> -keypass <currentPass> -new <newPass> -keystore C:\hp\UCMDB\UCMDBServer\conf\security\server.keystore

d. Enter the new password.

5. Start the UCMDB Server.

6. Repeat the procedure for the Server truststore.

How to Enable or Disable HTTP/HTTPS Ports

To enable or disable the HTTP/HTTPS ports from the JMX console:

1. Launch a Web browser and enter the following address: **https://localhost.<domain_ name>:8443/jmx-console.**

2. Enter the JMX console authentication credentials. (The default user name is **sysadmin**.)

3. Locate **UCMDB:service=Ports Management Services** and click the link to open the Operations page.

4. To enable or disable the HTTP port, locate the **HTTPSetEnable** operation and set the value.

   ○ **True:** the port is enabled.

   ○ **False:** the port is disabled.

5. To enable or disable the HTTPS port, locate the **HTTPSSetEnable** operation and set the value.

   ○ **True:** the port is enabled.

   ○ **False:** the port is disabled.

6. To enable or disable the HTTPS port with client authentication, locate the **HTTPSClientAuthSetEnable** operation and set the value.

   ○ **True:** the port is enabled.

   ○ **False:** the port is disabled.

How to Map the UCMDB Web Components to Ports

You can configure the mapping of each UCMDB component to the available ports from the JMX console.

To view the current component configurations:

1. Launch a Web browser and enter the following address: **https://localhost:8443/jmx-console**.

2. Enter the JMX console authentication credentials.

3. Locate **UCMDB:service=Ports Management Services** and click the link to open the Operations page.

4. Locate the **ComponentsConfigurations** method and click **Invoke**.

5. For each component, the valid ports and current mapped ports are displayed.

To map the components:

1. Locate **UCMDB:service=Ports Management Services** and click the link to open the Operations page.

2. Locate the **mapComponentToConnectors** method.

3. Enter a component name in the Value box. Select **True** or **False** for each of the ports corresponding to your selection. Click **Invoke**. The selected component is mapped to the selected ports. You can find the component names by invoking the **serverComponentsNames** method.

4. Repeat the process for each relevant component.

> **Note:**
>
> - Every component must be mapped to at least one port. If you do not map a component to any port, it is mapped by default to the HTTP port.
>
> - If you map a component to both the HTTPS port and the HTTPS port with client authentication, only the client authentication option is mapped (the other option is redundant in this case).
>
> - If you set **isHTTPSWithClientAuth** to **True** for the UCMDB UI component, you must also set it to **True** for the root component.

You can also change the value assigned to each of the ports.

To set values for the ports:

1. Locate **UCMDB:service=Ports Management Services** and click the link to open the Operations page.

2. To set a value for the HTTP port, locate the **HTTPSetPort** method and enter a value in the **Value** box. Click **Invoke**.

3. To set a value for the HTTPS port, locate the **HTTPSSetPort** method and enter a value in the **Value** box. Click **Invoke**.

4. To set a value for the HTTPS port with client authentication, locate the **HTTPSClientAuthSetPort** method and enter a value in the **Value** box. Click **Invoke**.

How to Modify the PostgreSQL Database Encrypted Password

This section explains how to modify the encrypted password for the PostgreSQL database user.

1. Create the Encrypted Form of a Password (AES, 192-bit key)

   > **Note:** In FIPS mode, it is 256-bit key.

   a. Access the Data Flow Probe JMX console. Launch a Web browser and enter the following address: **http://<Data Flow Probe machine name or IP address>:1977**. If you are running the Data Flow Probe locally, enter **http://localhost:1977**.

      You may have to log in with a user name and password.

   b. Locate the **Type=MainProbe** service and click the link to open the Operations page.

   c. Locate the **getEncryptedDBPassword** operation.

   d. In the **DB Password** field, enter the password to be encrypted.

   e. Invoke the operation by clicking the **getEncryptedDBPassword** button.

      The result of the invocation is an encrypted password string, for example:

      66,85,54,78,69,117,56,65,99,90,86,117,97,75,50,112,65,53,67,114,112,65,61,61

2. Stop the Data Flow Probe

   Start > All Programs > HP UCMDB > Stop Data Flow Probe

3. Run the set_dbuser_password.cmd Script

   This script is located in the following folder:
   C:\hp\UCMDB\DataFlowProbe\tools\dbscripts\set_dbuser_password.cmd

   Run the **set_dbuser_password.cmd** script with the new password as the first argument,and the PostgreSQL Root Account password as the second argument.

   For example:

   set_dbuser_password <my_password><root_password>.

   The password must be entered in its unencrypted form (as plain text).

4. Update the Password in the Data Flow Probe Configuration Files

a. The password must reside encrypted in the configuration files. To retrieve the password's encrypted form, use the **getEncryptedDBPassword** JMX method, as explained in step 1.

b. Add the encrypted password to the following properties in the **C:\hp\UCMDB\DataFlowProbe\conf\DataFlowProbe.properties** file.

- **appilog.agent.probe.jdbc.pwd**

  For example:

  ```
  appilog.agent.probe.jdbc.user = mamprobe
  appilog.agent.probe.jdbc.pwd =
  66,85,54,78,69,117,56,65,99,90,86,117,97,75,50,112,65,53,67,114,112,65,61
  ,61
  ```

- **appilog.agent.local.jdbc.pwd**

- **appilog.agent.normalization.jdbc.pwd**

5. Start the Data Flow Probe

   Start > All Programs > HP UCMDB > Start Data Flow Probe

How to Set the JMX Console Encrypted Password

This section explains how to encrypt the password for the JMX user. The encrypted password is stored in the DataFlowProbe.properties file. Users must log in to access the JMX console.

1. Create the Encrypted Form of a Password (AES, 192-bit key)

   Note: In FIPS mode, it is **256-bit** key.

   a. Access the Data Flow Probe JMX console. Launch a Web browser and enter the following address: **http://<Data Flow Probe machine name or IP address>:1977**. If you are running the Data Flow Probe locally, enter **http://localhost:1977**.

      You may have to log in with a user name and password.

   b. Locate the **Type=MainProbe** service and click the link to open the Operations page.

   c. Locate the **getEncryptedKeyPassword** operation.

   d. In the **Key Password** field, enter the password to be encrypted.

   e. Invoke the operation by clicking the **getEncryptedKeyPassword** button.

The result of the invocation is an encrypted password string, for example:

```
85,-9,-61,11,105,-93,-81,118
```

2. Stop the Data Flow Probe

Start > All Programs > HP UCMDB > Stop Data Flow Probe

3. Add the Encrypted Password

Add the encrypted password to the following property in the
C:\hp\UCMDB\DataFlowProbe\conf\DataFlowProbe.properties file.

appilog.agent.Probe.JMX.BasicAuth.Pwd

For example:

```
appilog.agent.Probe.JMX.BasicAuth.User=sysadmin
appilog.agent.Probe.JMX.BasicAuth.Pwd=12,-35,-37,82,-2,20,57,-40,38,80,-111,-
99,-64,-5,35,-122
```

> **Note:** To disable authentication, leaves these fields empty. If you do so, users can open the main page of the Probe's JMX console without entering authentication.

4. Start the Data Flow Probe

Start > All Programs > HP UCMDB > Start Data Flow Probe

Test the result in a Web browser.

How to Set the UpLoadScanFile Password

This section explains how to set the password for **UpLoadScanFile**, used for off-site scan saving. The encrypted password is stored in the **DataFlowProbe.properties** file. Users must log in to access the JMX console.

1. **Create the Encrypted Form of a Password (AES, 192-bit key)**

   a. Access the Data Flow Probe JMX console. Launch a Web browser and enter the following address: **http://<Data Flow Probe machine name or IP address>:1977**. If you are running the Data Flow Probe locally, enter **http://localhost:1977**.

You may have to log in with a user name and password.

 b. Locate the **Type=MainProbe** service and click the link to open the Operations page.

 c. Locate the **getEncryptedKeyPassword** operation.

 d. In the **Key Password** field, enter the password to be encrypted.

 e. Invoke the operation by clicking the **getEncryptedKeyPassword** button.

  The result of the invocation is an encrypted password string, for example:

```
85,-9,-61,11,105,-93,-81,118
```

2. **Stop the Data Flow Probe**

   Start > All Programs > HP UCMDB > Stop Data Flow Probe

3. **Add the Encrypted Password**

   Add the encrypted password to the following property in the
   C:\hp\UCMDB\DataFlowProbe\conf\DataFlowProbe.properties file.

   com.hp.ucmdb.discovery.Probe.JMX.UploadAuth.Pwd

   For example:

```
com.hp.ucmdb.discovery.Probe.JMX.UploadAuth.User=UploadScanFile
com.hp.ucmdb.discovery.Probe.JMX.UploadAuth.Pwd=116,116,21,34,-59,77,-
108,14,127,4,-89,101,-33,-31,116,53
```

4. **Start the Data Flow Probe**

   Start > All Programs > HP UCMDB > Start Data Flow Probe

   Test the result in a Web browser.

How to Retrieve the Current LW-SSO Configuration in a Distributed Environment

When UCMDB is embedded in a distributed environment, for example, in a BSM deployment, perform the following procedure to retrieve the current LW-SSO configuration on the processing machine.

To retrieve the current LW-SSO configuration:

1. Launch a Web browser and enter the following address: **https://localhost:8443/jmx-console.**

   You may be asked for a user name and password.

2. Locate **UCMDB:service=Security Services** and click the link to open the Operations page.

3. Locate the **retrieveLWSSOConfiguration** operation.

4. Click **Invoke** to retrieve the configuration.

How to Configure LW-SSO Settings

This procedure describes how to change the LW-SSO init string on the UCMDB server. This change is automatically sent to Probes (as an encrypted string), unless the UCMDB server is configured to not automatically do this. For details, see Disable Automatic Synchronization of the Confidential Manager Client Authentication and Encryption Settings Between the Server and Probes in the *HPE Universal CMDB Hardening Guide*.

1. On the UCMDB server, launch the Web browser and enter the following address: **https://localhost:8443/jmx-console**.

2. Click **UCMDB-UI:name=LW-SSO Configuration** to open the JMX MBEAN View page.

3. Locate the **setInitString** method.

4. Enter a new LW-SSO init string.

5. Click **Invoke**.

How to Configure Confidential Manager Communication Encryption

This procedure describes how to change the Confidential Manager communication encryption settings on the UCMDB Server. These settings specify how the communication between the Confidential Manager client and the Confidential Manager server is encrypted. This change is automatically sent to Probes (as an encrypted string), unless the UCMDB server is configured to not automatically do this. For details, see Disable Automatic Synchronization of the Confidential Manager Client Authentication and Encryption Settings Between the Server and Probes in the *HPE Universal CMDB Hardening Guide*.

1. On the UCMDB server, launch the Web browser and enter the following address: **https://localhost:8443/jmx-console.**

2. Click **UCMDB:service=Security Services** to open the JMX MBEAN View page.

3. Click the **CMGetConfiguration** method.

4. Click **Invoke**.

   The XML of the current Confidential Manager configuration is displayed.

5. Copy the contents of the displayed XML.

6. Navigate back to the **Security Services** JMX MBean View page.

7. Click the **CMSetConfiguration** method.

8. Paste the copied XML into the **Value** field.

9. Update the relevant transport-related settings and click **Invoke**.

   Example:

```
<transport>

      <encryptTransportMode>true</encryptTransportMode>

      <CMEncryptionDecryption>

            <encryptDecryptInitString>radiohead</encryptDecryptInitString>

            <cryptoSource>lw</cryptoSource>

            <lwJCEPBECompatibilityMode>true</lwJCEPBECompatibilityMode>

            <cipherType>symmetricBlockCipher</cipherType>

            <engineName>AES</engineName>

            <algorithmModeName>CBC</algorithmModeName>

            <algorithmPaddingName>PKCS7Padding</algorithmPaddingName>

            <keySize>256</keySize>

            <pbeCount>20</pbeCount>

            <pbeDigestAlgorithm>SHA1</pbeDigestAlgorithm>

            <encodingMode>Base64Url</encodingMode>

            <useMacWithCrypto>false</useMacWithCrypto>

            <macType>hmac</macType>

            <macKeySize>256</macKeySize>

            <macHashName>SHA256</macHashName>

      </CMEncryptionDecryption>

</transport>
```

For details about the values that can be updated, see Confidential Manager Encryption Settings in the *HPE Universal CMDB Hardening Guide*.

How to Configure Confidential Manager Client Authentication and Encryption Settings on the Probe

This procedure is relevant if the UCMDB Server has been configured to not send LW-SSO/Confidential Manager configuration and settings automatically to Probes. For details, see Disable Automatic Synchronization of the Confidential Manager Client Authentication and Encryption Settings Between the Server and Probes in the *HPE Universal CMDB Hardening Guide*.

1. On the Probe machine, launch the Web browser and enter the following address: **http://localhost:1977**.

   Note: If the Probe Manager and the Probe Gateway are running as separate processes, the address should be entered on the machine that is running the Probe Manager as follows: **http://localhost:1978**.

2. Click **type=CMClient** to open the JMX MBEAN View page.

3. Locate the **setLWSSOInitString** method and provide the same init string that was provided for UCMDB's LW-SSO configuration.

4. Click the **setLWSSOInitString** button.

How to Configure Confidential Manager Communication Encryption on the Probe

This procedure is relevant if the UCMDB Server has been configured to not send LW-SSO/Confidential Manager configuration and settings automatically to Probes.

1. On the Probe machine, launch the Web browser and enter the following address: **http://localhost:1977**.

   Note: If the Probe Manager and the Probe Gateway are running as separate processes, the address should be entered on the machine that is running the Probe Manager as follows: **http://localhost:1978**.

2. Click **type=CMClient** to open the JMX MBEAN View page.

3. Update the following transport-related settings:

> **Note:** You must update the same settings that you updated on the UCMDB server. To do this, some of the methods that you update on the Probe may require more than one parameter.

   a. **setTransportInitString** changes the **encryptDecryptInitString** setting.

   b. **setTransportEncryptionAlgorithm** changes Confidential Manager settings on the Probe according to the following map:

   - **Engine name** refers to the <engineName> entry

   - **Key size** refers to the <keySize> entry

   - **Algorithm padding name** refers to the <algorithmPaddingName> entry

   - **PBE count** refers to the <pbeCount> entry

   - **PBE digest algorithm** refers to the <pbeDigestAlgorithm> entry

   c. **setTransportEncryptionLibrary** changes Confidential Manager settings on the Probe according to the following map:

   - **Encryption Library name** refers to the <cryptoSource> entry

   - **Support previous lightweight cryptography versions** refers to the <lwJCEPBECompatibilityMode> entry

   d. **setTransportMacDetails** change Confidential Manager settings on the Probe according to the following map:

   - **Use MAC with cryptography** refers to the <useMacWithCrypto> entry

   - **MAC key size** refers to the <macKeySize> entry

4. Click the **reloadTransportConfiguration** button to make the changes effective on the Probe.

How to Configure the Confidential Manager Client's Cache Encryption Settings on the Probe

This procedure describes how to change the encryption settings of the Confidential Manager client's file system cache file. Note that changing the encryption settings for the Confidential Manager client's file system cache causes the file system cache file to be recreated. This recreation process requires restarting the Probe and full synchronization with the UCMDB Server.

1. On the Probe machine, launch the Web browser and enter the following address:
   http://localhost:1977.

   Note: If the Probe Manager and the Probe Gateway are running as separate processes, the
   address should be entered on the machine that is running the Probe Manager as follows:
   http://localhost:1978.

2. Click **type=CMClient** to open the JMX MBEAN View page.

3. Update the following cache-related settings:

   Note: Some of the methods that you update on the Probe may require more than one
   parameter. To see the current probe configuration, click **displayCacheConfiguration** in the
   JMX MBEAN View page.

   a. **setCacheInitString** changes the file system cache <encryptDecryptInitString> setting.

   b. **setCacheEncryptionAlgorithm** changes the file system cache settings according to the
   following map:

      - **Engine name** refers to the <engineName> entry

      - **Key size** refers to the <keySize> entry

      - **Algorithm padding name** refers to the <algorithmPaddingName> entry

      - **PBE count** refers to the <pbeCount> entry

      - **PBE digest algorithm** refers to the <pbeDigestAlgorithm> entry

   c. **setCacheEncryptionLibrary** changes the cache file system settings according to the
   following map:

      - **Encryption Library name** refers to the <cryptoSource> entry

      - **Support previous lightweight cryptography versions** refers to the
        <lwJCEPBECompatibilityMode> entry

   d. **setCacheMacDetails** changes the cache file system settings according to the following map:

      - **Use MAC with cryptography** refers to the <useMacWithCrypto> entry

      - **MAC key size** refers to the <macKeySize> entry

4. Click the **reloadCacheConfiguration** button to make the changes effective on the Probe. This
   causes the Probe to restart.

   Note: Make sure that no job is running on the Probe during this action.

How to Export and Import Credential and Range Information in Encrypted Format

You can export and import credentials and network range information in encrypted format in order to copy the credentials information from one UCMDB Server to another. For example, you might perform this operation during recovery following a system crash or during upgrade.

- **When exporting credentials information,** you must enter a password (of your choosing). The information is encrypted with this password.

- **When importing credentials information,** you must use the same password that was defined when the DSD file was exported.

Note: The exported credentials document also contains ranges information that is defined on the system from which the document was exported. During the import of the credentials document, ranges information is imported as well.

To export credentials information from the UCMDB Server:

1. On the UCMDB Server, launch the Web browser and enter the following address: **https://localhost:8443/jmx-console**. You may have to log in with a user name and password.

2. Click **UCMDB:service=DiscoveryManager** to open the JMX MBEAN View page.

3. Locate the **exportCredentialsAndRangesInformation** operation. Do the following:

   - Enter your customer ID (the default is 1).

   - Enter a name for the exported file.

   - Enter your password.

   - Set **isEncrypted=True** if you want the exported file to be encrypted with the provided password, or **isEncrypted=False** if you want the exported file to not be encrypted (in which case passwords and other sensitive information are not exported).

4. Click **Invoke** to export.

   When the export process completes successfully, the file is saved to the following location: **c:\hp\UCMDB\UCMDBServer\conf\discovery\<customer_dir>**.

To import credentials information from the UCMDB Server:

1. On the UCMDB Server, launch the Web browser and enter the following address: **https://localhost:8443/jmx-console**.

   You may have to log in with a user name and password.

2. Click **UCMDB:service=DiscoveryManager** to open the JMX MBEAN View page.

3. Locate the **importCredentialsAndRangesInformation** operation.

4. Enter your customer ID (the default is 1).

5. Enter the name of the file to import. This file must be located in **c:\hp\UCMDB\UCMDBServer\conf\discovery\<customer_dir>**.

6. Enter the password. This must be the same password that was used when the file was exported.

7. Click **Invoke** to import the credentials.

How to Remove Credential and Range Information by Domain Name

This task describes how to remove credential and range information for a selected domain.

1. On the UCMDB server, launch the Web browser and navigate to: **https://localhost:8443/jmx-console**. You may have to log in with a user name and password.

2. Locate **UCMDB:service=Discovery Manager** and click the link to jump to the Operations table.

3. Locate the **cleanCredentialsAndRangesInformation** operation.

4. In the **Value** field for **customerID**, enter your customer ID.

5. In the **Value** field for **domainName**, enter the name of the domain that you want to remove.

6. Click **Invoke**.

How to Generate or Update the Encryption Key for Confidential Manager

You can generate or update an encryption key to be used for encryption or decryption of Confidential Manager communication and authentication configurations exchanged between the UCMDB Server and the Data Flow Probe. In each case (generate or update), the UCMDB Server creates a new encryption key based on parameters that you supply (for example, key length, extra PBE cycles, JCE provider) and distributes it to the Probes.

The result of running the **generateEncryptionKey** method is a new generated encryption key. This key is stored only in secured storage and its name and details are not known. If you reinstall an existing Data Flow Probe, or connect a new Probe to the UCMDB Server, this new generated key is not recognized by the new Probe. In these cases, it is preferable to use the **changeEncryptionKey** method to change encryption keys. This way, when you reinstall a Probe or install a new Probe, you can import the existing key (whose name and location you know) by running the **importEncryptionKey** method on the Probe JMX console.

Note:

- The difference between the methods used to create a key (**generateEncryptionKey**) and update a key (**changeEncryptionKey**) is that **generateEncryptionKey** creates a new, random encryption key, while **changeEncryptionKey** imports an encryption key whose name you provide.

- Only one encryption key can exist on a system, no matter how many Probes are installed.

This task includes the following steps:

- "Generate a New Encryption Key" below

- "Update an Encryption Key on a UCMDB Server" on the next page

- "Update an Encryption Key on a Probe" on page 128

- "Manually Change the Encryption Key when the Probe Manager and Probe Gateway are Installed on Separate Machines" on page 129

- "Define Several JCE Providers" on page 129

Generate a New Encryption Key

You can generate a new key to be used by the UCMDB Server and Data Flow Probe for encryption or decryption. The UCMDB Server replaces the old key with the new generated key, and distributes this key among the Probes.

To generate a new encryption key through the JMX console:

1. On the UCMDB server, launch the Web browser and enter the following address: **https://localhost:8443/jmx-console**.

   You may have to log in with a user name and password.

2. Click **UCMDB:service=DiscoveryManager** to open the JMX MBEAN View page.

3. Locate the generateEncryptionKey operation.

a. In the **customerId** parameter box, enter 1 (the default).

b. For **keySize**, specify the length of the encryption key. Valid values are 128, 192, or 256.

c. For **usePBE**, specify **True** or **False**:

   - **True:** use additional PBE hash cycles.

   - **False:** do not use additional PBE hash cycles.

d. For **jceVendor**, you can choose to use a non-default JCE provider. If the box is empty, the default provider is used.

e. For **autoUpdateProbe**, specify **True** or **False**:

   - **True:** the server distributes the new key to the Probes automatically.

   - **False:** the new key should be placed on the Probes manually.

f. For **exportEncryptionKey**, specify **True** or **False**.

   - **True:** In addition to creating the new password and storing it in secured storage, the Server exports the new password to the file system (**c:\hp\UCMDB\UCMDBServer\conf\discovery\key.bin**). This option enables you to update Probes manually with the new password.

   - **False:** The new password is not exported to the file system. To update Probes manually, set **autoUpdateProbe** to False and **exportEncryptionKey** to True.

   > **Caution:** Make sure that the Probe is up and connected to the server. If the Probe goes down, the key cannot reach the Probe. If you change the key before the Probe goes down, once the Probe is up again, the key is sent again to the Probe. However, if you have changed the key more than once before the Probe goes down, you must change the key manually through the JMX console. (Select **False** for **exportEncryptionKey**).

4. Click **Invoke** to generate the encryption key.

Update an Encryption Key on a UCMDB Server

You use the **changeEncryptionKey** method to import your own encryption key to the UCMDB server and distribute it among all Probes.

To update an encryption key through the JMX Console:

1. Copy the `key.bin` file you generated in "Generate a New Encryption Key" on the previous page to the **C:\hp\UCMDB\UCMDBServer\conf\discovery\customer_1** directory, and rename the

key.bin file. For example, key_1.bin.

> **Note:** Make sure you rename the key.bin file.

2. On the UCMDB Server, launch the Web browser and enter the following address: **https://localhost:8443/jmx-console.** You may have to log in with a user name and password.

3. Click **UCMDB:service=DiscoveryManager** to open the JMX MBEAN View page.

4. Locate the **changeEncryptionKey** operation.

   a. In the **customerId** parameter box, enter **1** (the default).

   b. For **newKeyFileName**, enter the name of the new key.

   c. For **keySizeInBits**, specify the length of the encryption key. Valid values are 128, 192, or 256.

   d. For **usePBE**, specify **True** or **False**:

      - **True:** use additional PBE hash cycles.

      - **False:** do not use additional PBE hash cycles.

   e. For **jceVendor**, you can choose to use a non-default JCE provider. If the box is empty, the default provider is used.

   f. For **autoUpdateProbe**, specify **True** or **False**:

      - **True:** the server distributes the new key to the Probes automatically.

      - **False:** the new key should be distributed manually using the Probe JMX console.

      > **Caution:** Make sure that the Probe is up and connected to the server. If the Probe goes down, the key cannot reach the Probe. If you change the key before the Probe goes down, once the Probe is up again, the key is sent again to the Probe. However, if you have changed the key more than once before the Probe goes down, you must change the key manually through the JMX console. (Select **False** for **autoUpdateProbe**).

5. Click **Invoke** to generate and update the encryption key.

Update an Encryption Key on a Probe

If you choose not to distribute an encryption key from the UCMDB Server to all Probes automatically (because of security concerns), you should download the new encryption key to all Probes and run the **importEncryptionKey** method on the Probe:

1. Place the encryption key file in **C:\hp\UCMDB\DataFlowProbe\conf\security**.

2. On the Probe machine, launch the Web browser and enter the following address: **http://localhost:1977**.

   You may have to log in with a user name and password.

   > **Note:** If the Probe Manager and the Probe Gateway are running as separate processes, the address should be entered on the machine that is running the Probe Manager as follows: **http://localhost:1978**.

3. On the Probe domain, click **type=SecurityManagerService**.

4. Locate the **importEncryptionKey** method.

5. Enter the name of the encryption key file that resides in **C:\hp\UCMDB\DataFlowProbe\conf\security**. This file contains the key to be imported.

6. Click the **importEncryptionKey** button.

7. Perform a restart of the probe.

Manually Change the Encryption Key when the Probe Manager and Probe Gateway are Installed on Separate Machines

1. On the Probe Manager machine, start the Probe Manager service (**Start > Programs > HP UCMDB > Probe Manager**).

2. Import the key from the server, using the Probe Manager JMX. For details, see "Generate a New Encryption Key" on page 126.

3. After the encryption key is imported successfully, restart the Probe Manager and Probe Gateway services.

Define Several JCE Providers

When you generate an encryption key through the JMX Console, you can define several JCE providers, using the **changeEncryptionKey** and **generateEncryptionKey** methods.

To change the default JCE provider:

1. Register the JCE provider jar files in **$JRE_HOME/lib/ext**.

2. Copy the jar files to the $JRE_HOME folder:

   ○ For the UCMDB Server: $JRE_HOME resides at: **c:\hp\UCMDB\UCMDBServer\bin\jre**

   ○ For the Data Flow Probe: $JRE_HOME resides at: **c:\hp\UCMDB\DataFlowProbe\bin\jre**

3. Add the provider class at the end of the provider list in the **$JRE_HOME\lib\security\java.security** file.

4. Update the **local_policy.jar** and **US_export_policy.jar** files to include unlimited JCE policies. You can download these jar files from the Sun website.

5. Restart the UCMDB Server and the Data Flow Probe.

6. Locate the JCE vendor field for the **changeEncryptionKey** or **generateEncryptionKey** method, and add the name of the JCE provider.

How to Configure CAC Support on UCMDB

This section describes how to configure Smart Card Authentication or PKI Authentication (CAC) support on UCMDB.

> **Note:** CAC support is only available when using Internet Explorer 10 or later.

1. Import the root CA and any intermediate certificates into the UCMDB Server Truststore as follows:

   a. On the UCMDB machine, copy the certificate files to the following directory on UCMDB:

   C:\HP\UCMDB\UCMDBServer\conf\security

   > **Note:** If your certificate is in Microsoft p7b format, you may need to convert it to PEM format.

   b. For each certificate, run the following command:

   ```
   C:\HP\UCMDB\UCMDBServer\bin\jre\bin\keytool.exe -import -v -keystore
   C:\HP\UCMDB\UCMDBServer\conf\security\server.truststore -file
   <certificate> -alias <certificate alias>
   ```

   c. Enter the UCMDB Server Truststore password (default **hppass**).

d. When asked, **Trust this certificate?**, press **y** and then **Enter**.

e. Make sure the output **Certificate** was added to the keystore.

2. Open the JMX console by launching the Web browser and entering the Server address, as follows: **https://<UCMDB Server Host Name or IP>:8443/jmx-console**.

You may have to log in with a user name and password.

3. Under UCMDB, click **UCMDB:service=Ports Management Services** to open the Operations page.

   ○ (optional) Click **ComponentsConfigurations**. Do the following:

      • Set **HTTPSClientAuthSetPort** to **8444** and click **Invoke**.

      • Click **Back to MBean**.

   ○ Click **mapComponentToConnectors**. Do the following:

      • In the mapComponentToConnectors service, set **componentName** to **ucmdb-ui**.

      • Set only **isHTTPSWithClientAuth** to **true**, and click **Invoke**.

      • Click **Back to MBean**.

      • In the mapComponentToConnectors service, set **componentName** to **root**.

      • Set only **isHTTPSWithClientAuth** to **true**, and click **Invoke**.

4. Under UCMDB, click **UCMDB:service=Security Services** to open the Operations page. In the **loginWithCAC** service, do the following:

   ○ Set **loginWithCAC** to **true**, and click **Invoke**.

      Note: If a user who is used in CAC login does not have permissions to access the UCMDB UI, then automatic login and display of a white page will not occur.

   ○ Click **Back to MBean**.

   ○ (optional) Click **usernameField** to specify the field from the certificate that will be used by UCMDB to extract a username, and click **Invoke**.

      Note: If you do not specify a field, the default of PRINCIPAL_NAME_FROM_SAN_ FIELD is used.

   ○ Click **Back to MBean**.

   ○ Click **pathToCRL** to set a path to an offline Certificate Revocation List (CRL) to be used if the online list (from the certificate) is not available, and click **Invoke**.

> **Note:** When you work with a local CRL and there is a working Internet connection to the UCMDB server, the local CRL is used. The validation of any certificate (even if it is not revoked) fails in the following situations:
>
> - if the CRL path is set but the CRL file itself is missing
>
> - if the CRL is expired
>
> - if the CRL has an incorrect signature
>
> If you do not set the path to an offline CRL and the UCMDB server cannot access the online CRL, all certificates that contain a CRL or OCSP URL are rejected (since the URL cannot be accessed, the revocation check fails). To give the UCMDB server access to the Internet, uncomment the following lines in the **wrapper.conf** file and provide a valid proxy and port:
>
> ```
> #wrapper.java.additional.40=-Dhttp.proxyHost=<PROXY_ADDR>
> #wrapper.java.additional.41=-Dhttp.proxyPort=<PORT>
> #wrapper.java.additional.42=-Dhttps.proxyHost=<PROXY_ADDR>
> #wrapper.java.additional.43=-Dhttps.proxyPort=<PORT>
> ```

- Click **Back to MBean**.

- (optional) Set **onlyCACCerts** to **true**, and click **Invoke**.

   Set this operation to **true** to accept only certificates that come from a physical CAC device.

   You should now be able to log into UCMDB with **https://<UCMDB Server Host Name or IP>.<domainname>:8444**.

5. Configure UCMDB to use LW-SSO authentication and restart the UCMDB Server.

For details on LW-SSO authentication, see "Enabling Login to HP Universal CMDB with LW-SSO" in the *HPE Universal CMDB Hardening Guide*.

How to Configure CAC Support for UCMDB by Reverse Proxy

This section describes how to configure Common Access Card (CAC) support on UCMDB using a reverse proxy.

1. Open the JMX console by launching the Web browser and entering the Server address, as follows: **https://localhost:8443/jmx-console**.

   You may have to log in with a user name and password.

2. Under UCMDB, click **UCMDB:service=Ports Management Services** to open the Operations page.

   ○ (optional) Click **ComponentsConfigurations**. Do the following:

     • Set **HTTPSetPort** to **8080** and click **Invoke**.

     • Click **Back to MBean**.

   ○ Click **mapComponentToConnectors**. Do the following:

     • In the mapComponentToConnectors service, set **componentName** to **ucmdb-ui**.

     • Set only **isHTTP** to **true**, and click **Invoke**.

     • Click **Back to MBean**.

     • In the mapComponentToConnectors service, set **componentName** to **root**.

     • Set only **isHTTP** to **true**, and click **Invoke**.

3. Under UCMDB, click **UCMDB:service=Security Services** to open the Operations page.

   ○ Set **loginWithCAC** to **true**, and click **Invoke**.

   ○ Click **Back to MBean**.

   ○ Set **withReverseProxy** to **true**, and click **Invoke**.

     This setting tells the UCMDB server to extract from the UCMDB_SSL_CLIENT_CERT header the user name to be used in UCMDB and the certificate to be used for authentication.

   ○ Click **Back to MBean**.

   ○ (optional) Set **onlyCACCerts** to **true**, and click **Invoke**.

     Set this operation to **true** to accept only certificates that come from a physical CAC device.

   ○ (optional) Click **usernameField** to specify the field from the certificate that will be used by UCMDB to extract a username, and click **Invoke**.

     Note: If you do not specify a field, the default of PRINCIPAL_NAME_FROM_SAN_ FIELD is used.

4. Restart the UCMDB Server.

(Optional) Configure LocationMatch in Apache Reverse Proxy httpd-ssl.conf for CAC Setup

When using Apache as a reverse proxy while accessing the UCMDB server with CAC enabled, sometimes you might encounter cascading pin request popups.

To skip the **appler.jsp** pin prompt, you may add the **LocationMatch** setting as follows to the **httpd-ssl.conf** file:

```
<LocationMatch "^/ucmdb-ui/login_page.jsp">
    SSLVerifyClient require
    SSLVerifyDepth  10
</LocationMatch>
```

Example: Apache 2.4.4 Configuration

This section describes a sample configuration file for Apache 2.4.4.

> **Note:** This example presumes that the Apache server was installed in **c:\Apache24**; if it is installed in a different folder, you must change the example in all cases to specify the correct location.

The port for mutual authentication used in this example is 443. In the **c:\Apache24\conf** folder, copy the following:

- the certificate used by the apache server (**server.crt**)

- the private key of the Apache server (**server.key**)

- the trusted CAs of the Apache server (**ssl.crt**)

- the certification revocation list (**ssl.crt**).

> **Note:** These four files must all be in PEM format.

Replace the content of **c:\Apache24\conf\httpd.conf** with the following (change the [APACHE_MACHINE_FQD] accordingly):

```
ServerRoot "c:/Apache24"
Listen 80
LoadModule access_compat_module modules/mod_access_compat.so
LoadModule actions_module modules/mod_actions.so
LoadModule alias_module modules/mod_alias.so
LoadModule allowmethods_module modules/mod_allowmethods.so
LoadModule asis_module modules/mod_asis.so
LoadModule auth_basic_module modules/mod_auth_basic.so
LoadModule authn_core_module modules/mod_authn_core.so
LoadModule authn_file_module modules/mod_authn_file.so
LoadModule authz_core_module modules/mod_authz_core.so
LoadModule authz_groupfile_module modules/mod_authz_groupfile.so
LoadModule authz_host_module modules/mod_authz_host.so
```

```
LoadModule authz_user_module modules/mod_authz_user.so
LoadModule autoindex_module modules/mod_autoindex.so
LoadModule cgi_module modules/mod_cgi.so
LoadModule dir_module modules/mod_dir.so
LoadModule env_module modules/mod_env.so
LoadModule headers_module modules/mod_headers.so
LoadModule include_module modules/mod_include.so
LoadModule isapi_module modules/mod_isapi.so
LoadModule log_config_module modules/mod_log_config.so
LoadModule mime_module modules/mod_mime.so
LoadModule negotiation_module modules/mod_negotiation.so
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_html_module modules/mod_proxy_html.so
LoadModule proxy_http_module modules/mod_proxy_http.so
LoadModule setenvif_module modules/mod_setenvif.so
LoadModule socache_shmcb_module modules/mod_socache_shmcb.so
LoadModule ssl_module modules/mod_ssl.so
LoadModule xml2enc_module modules/mod_xml2enc.so
<IfModule unixd_module>
User daemon
Group daemon
</IfModule>
ServerAdmin admin@example.com
ServerName [APACHE_MACHINE_FQD]:80
<Directory />
    AllowOverride none
    Require all denied
</Directory>
DocumentRoot "c:/Apache24/htdocs"
<Directory "c:/Apache24/htdocs">
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>
<IfModule dir_module>
    DirectoryIndex index.html
</IfModule>
<Files ".ht*">
    Require all denied
</Files>
ErrorLog "logs/error.log"
LogLevel warn
<IfModule log_config_module>
    LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\""
combined
    LogFormat "%h %l %u %t \"%r\" %>s %b" common
    <IfModule logio_module>
      LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\" %I
```

```
%O" combinedio
    </IfModule>
    CustomLog "logs/access.log" common
</IfModule>
<IfModule alias_module>
    ScriptAlias /cgi-bin/ "c:/Apache24/cgi-bin/"
</IfModule>
<IfModule cgid_module>
</IfModule>
<Directory "c:/Apache24/cgi-bin">
    AllowOverride None
    Options None
    Require all granted
</Directory>
<IfModule mime_module>
    TypesConfig conf/mime.types
    AddType application/x-compress .Z
    AddType application/x-gzip .gz .tgz
</IfModule>
<IfModule proxy_html_module>
Include conf/extra/proxy-html.conf
</IfModule>
Include conf/extra/httpd-ssl.conf
<IfModule ssl_module>
SSLRandomSeed startup builtin
SSLRandomSeed connect builtin
</IfModule>
```

Also, replace the content of **c:\Apache24\conf\extra\httpd-ssl.conf** with the following (change the [APACHE_MACHINE_FQD] , [UCMDB_SERVER_NAME], and [UCMDB_CM_SERVER_NAME] accordingly):

```
Listen 443
SSLCipherSuite HIGH:MEDIUM:!aNULL:!MD5
SSLPassPhraseDialog  builtin
SSLSessionCache         "shmcb:c:/Apache24/logs/ssl_scache(512000)"
SSLSessionCacheTimeout  300
<VirtualHost _default_:443>
DocumentRoot "c:/Apache24/htdocs"
ServerName [APACHE_MACHINE_FQD]:443
ServerAdmin admin@example.com
ErrorLog "c:/Apache24/logs/error.log"
TransferLog "c:/Apache24/logs/access.log"
SSLEngine on
SSLCertificateFile "c:/Apache24/conf/server.crt"
SSLCertificateKeyFile "c:/Apache24/conf/server.key"
SSLCACertificateFile "c:/Apache24/conf/ssl.crt"
SSLCARevocationFile "c:/Apache24/conf/ssl.crl"
```

```
SSLCARevocationCheck leaf
SSLVerifyClient require
SSLVerifyDepth  10
SSLOptions +ExportCertData
<FilesMatch "\.(cgi|shtml|phtml|php)$">
    SSLOptions +StdEnvVars
</FilesMatch>
<Directory "c:/Apache24/cgi-bin">
    SSLOptions +StdEnvVars
</Directory>
BrowserMatch "MSIE [2-5]" \
        nokeepalive ssl-unclean-shutdown \
        downgrade-1.0 force-response-1.0
CustomLog "c:/Apache24/logs/ssl_request.log" \
        "%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x \"%r\" %b"
RequestHeader set UCMDB_SSL_CLIENT_CERT %{SSL_CLIENT_CERT}e
ProxyRequests off
<Proxy *>
    Order deny,allow
    Deny from all
    Allow from all
</Proxy>
ProxyPass / http://[UCMDB_SERVER_NAME]:8080/
ProxyPassReverse / http://[UCMDB_SERVER_NAME]:8080/
ProxyPass /mam http://[UCMDB_SERVER_NAME]:8080/mam
ProxyPassReverse /mam http://[UCMDB_SERVER_NAME]:8080/mam
ProxyPass /mam_images http://[UCMDB_SERVER_NAME]:8080/mam_images
ProxyPassReverse /mam_images http://[UCMDB_SERVER_NAME]:8080/mam_images
ProxyPass /mam-collectors http://[UCMDB_SERVER_NAME]:8080/mam-collectors
ProxyPassReverse /mam-collectors http://[UCMDB_SERVER_NAME]:8080/mam-collectors
ProxyPass /ucmdb http://[UCMDB_SERVER_NAME]:8080/ucmdb
ProxyPassReverse /ucmdb http://[UCMDB_SERVER_NAME]:8080/ucmdb
ProxyPass /site http://[UCMDB_SERVER_NAME]:8080/site
ProxyPassReverse /site http://[UCMDB_SERVER_NAME]:8080/site
ProxyPass /ucmdb-ui http://[UCMDB_SERVER_NAME]:8080/ucmdb-ui
ProxyPassReverse /ucmdb-ui http://[UCMDB_SERVER_NAME]:8080/ucmdb-ui
ProxyPass /status http://[UCMDB_SERVER_NAME]:8080/status
ProxyPassReverse /status http://[UCMDB_SERVER_NAME]:8080/status
ProxyPass /jmx-console http://[UCMDB_SERVER_NAME]:8443/jmx-console
ProxyPassReverse /jmx-console http://[UCMDB_SERVER_NAME]:8443/jmx-console
ProxyPass /axis2 http://[UCMDB_SERVER_NAME]:8080/axis2
ProxyPassReverse /axis2 http://[UCMDB_SERVER_NAME]:8080/axis2
ProxyPass /icons http://[UCMDB_SERVER_NAME]:8080/icons
ProxyPassReverse /icons http://[UCMDB_SERVER_NAME]:8080/icons
ProxyPass /ucmdb-api http://[UCMDB_SERVER_NAME]:8080/ucmdb-api
ProxyPassReverse /ucmdb-api http://[UCMDB_SERVER_NAME]:8080/ucmdb-api
ProxyPass /ucmdb-docs http://[UCMDB_SERVER_NAME]:8080/ucmdb-docs
ProxyPassReverse /ucmdb-docs http://[UCMDB_SERVER_NAME]:8080/ucmdb-docs
```

```
ProxyPass /ucmdb-api/8.0 http://[UCMDB_SERVER_NAME]:8080/ucmdb-api/8.0
ProxyPassReverse /ucmdb-api/8.0 http://[UCMDB_SERVER_NAME]:8080/ucmdb-api/8.0
ProxyPass /cm http://[UCMDB_SERVER_NAME]:8080/cm
ProxyPassReverse /cm http://[UCMDB_SERVER_NAME]:8080/cm
ProxyPass /cnc http://[UCMDB_CM_SERVER_NAME]/cnc
ProxyPassReverse /cnc http://[UCMDB_CM_SERVER_NAME]/cnc
ProxyPass /docs http://[UCMDB_CM_SERVER_NAME]/docs
ProxyPassReverse /docs http://[UCMDB_CM_SERVER_NAME]/docs
ProxyPass /ucmdb-browser http://[UCMDB_CM_SERVER_NAME]/ucmdb-browser
ProxyPassReverse /ucmdb-browser http://[UCMDB_CM_SERVER_NAME]/ucmdb-browser
</VirtualHost>
<LocationMatch "^/ucmdb-ui/login_page.jsp">
    SSLVerifyClient require
    SSLVerifyDepth  10
</LocationMatch>
```

Now you can access the UCMDB server through revers proxy by going to **https://[APACHE_ MACHINE_FQD]**.

> **Note:** You must have a valid certificate imported in Internet Explorer. A valid certificate is one that was signed by a CA of the Apache trusted CAs (it must be present in the **ssl.crt** file).

How to Harden the Data Flow Probe Connector in UCMDB

1. Access the UCMDB JMX console: In your Web browser, enter the following URL:
   https://<ucmdb machine name or IP address>:8443/jmx-console. You may have to log in with a user name and password.

2. Select the service: **Ports Management Services**.

3. Invoke the **PortsDetails** method, and note the port number for HTTPS. (Default: 8443) Ensure that the value in the **Is Enabled** column is **True**.

4. Return to **Ports Management Services**.

5. To map the Data Flow Probe connector to server authentication mode, invoke the **mapComponentToConnectors** method with the following parameters:

   ○ **componentName**: mam-collectors

   ○ **isHTTPS**: true

   ○ **All other flags**: false

The following message is displayed:

```
Operation succeeded. Component mam-collectors is now mapped to: HTTPS ports.
```

> **Note:** If you want to use multiple authentication methods, make sure you check the ports used by each of them and set them to **true** (when mapping both cm and mam-collectors).

6. Return to **Ports Management Services**.

7. To map the Confidential Manager connector to server authentication mode, invoke the **mapComponentToConnectors** method with the following parameters:

   ○ **componentName**: cm

   ○ **isHTTPS**: true

   ○ **All other flags**: false

   The following message is displayed:

```
Operation succeeded. Component cm is now mapped to: HTTPS ports.
```

> **Note:** If you want to use multiple authentication methods, make sure you check the ports used by each of them and set them to **true** (when mapping both cm and mam-collectors).

How to Encrypt the Probe Keystore and Truststore Passwords

The Probe keystore and truststore passwords are stored encrypted in **C:\HP\UCMDB\DataFlowProbe\conf\security\ssl.properties**. This procedure explains how to encrypt the password.

1. Start Data Flow Probe (or verify that it is already running).

2. Access the Data Flow Probe JMX console: Launch a Web browser and enter the following address: http://<Data Flow Probe machine name or IP address>:1977. If you are running the Data Flow Probe locally, enter **http://localhost:1977**.

   > **Note:** You may have to log in with a user name and password.

3. Locate the **Type=MainProbe** service and click the link to open the Operations page.

4. Locate the **getEncryptedKeyPassword** operation.

5. Enter your keystore or truststore password in the **Key Password** field and invoke the operation by clicking **getEncryptedKeyPassword**.

6. The result of the invocation is an encrypted password string, for example:

   66,85,54,78,69,117,56,65,99,90,86,117,97,75,50,112,65,53,67,114,112,65,61,61

7. Copy and paste the encrypted password into the line relevant to either the keystore or the truststore in the following file: **C:\HP\UCMDB\DataFlowProbe\conf\security\ssl.properties.**

How to Enable Login to HP Universal CMDB with LW-SSO

1. Access the JMX console by entering the following address into your Web browser: **https://<server_name>:8443/jmx-console**, where **<server_name>** is the name of the machine on which HP Universal CMDB is installed.

2. Under **UCMDB-UI**, click the **name=LW-SSO Configuration** to open the Operations page.

3. Set the init string using the **setInitString** method.

4. Set the domain name of the machine on which UCMDB is installed using the **setDomain** method.

5. Invoke the method **setEnabledForUI** with the parameter set to **True.**

6. **Optional**. If you want to work using multi-domain functionality, select the **addTrustedDomains** method, enter the domain values and click **Invoke**.

7. **Optional.** If you want to work using a reverse proxy, select the **updateReverseProxy** method, set the **Is reverse proxy enabled** parameter to **True**, enter a URL for the **Reverse proxy full server URL** parameter, and click Invoke. If you want to access UCMDB both directly and using a reverse proxy, set the following additional configuration: select the **setReverseProxyIPs** method, enter the IP address for the Reverse proxy ip/s parameter and click **Invoke.**

8. **Optional**. If you want to access UCMDB using an external authentication point, select the **setValidationPointHandlerEnable** method, set the **Is validation point handler enabled** parameter to **True**, enter the URL for the authentication point in the **Authentication point server** parameter, and click **Invoke.**

9. To view the LW-SSO configuration as it is saved in the settings mechanism, invoke the **retrieveConfigurationFromSettings** method.

10. To view the actual loaded LW-SSO configuration, invoke the **retrieveConfiguration** method.

> **Note:** You cannot enable LW-SSO via the user interface.

How to Test LDAP Connections

This section describes a method of testing the LDAP authentication configuration using the JMX console.

1. Launch your Web browser and enter the following address: **https://<server_name>:8443/jmx-console**, where **<server_name>** is the name of the machine on which HP Universal CMDB is installed.

   You may need to log in with a user name and password.

2. Under **UCMDB**, click **UCMDB:service=LDAP Services** to open the Operations page.

3. Locate **testLDAPConnection.**

4. Enter the customer ID in the **customer id** value box.

5. In the **Value** box for the **ldapHost** parameter, enter the ldap host.

6. Click **Invoke.**

   The JMX MBEAN Operation Result page indicates whether the LDAP connection is successful. If the connection is successful, the page also shows the LDAP root groups.

How to Enable and Define the LDAP Authentication Method

This task describes how to configure LDAP authentication settings using the JMX console.

> **Note:**
>
> - In a high availability environment, make sure you log in to the JMX console of the Writer server.
>
> - For an example of LDAP authentication settings, see "LDAP Authentication Settings - Example" in the *HPE Universal CMDB Hardening Guide*.

To configure single or multiple LDAP authentication settings:

1. Launch your Web browser and enter the following address: **https://<server_name>:8443/jmx-console**, where **<server_name>** is the name of the machine on which HP Universal CMDB is installed.

   You may need to log in with a user name and password.

2. Under **UCMDB,** click **UCMDB:service=LDAP Services** to open the Operations page.

3. To view the current LDAP authentication settings, locate the **getLDAPSettings** method. Click **Invoke**. A table displays all the LDAP settings and their values.

> **Note:** If you need to view the LDAP settings and their values for only one LDAP server, enter the LDAP server in the **ldapHost** field, and then click **Invoke**.

4. To configure a new LDAP server, locate the **configureLDAPServer** method. Enter the values for the relevant settings and click **Invoke**.

> **Note:**
>
> ○ You can specify the order in which the LDAP servers are presented in UCMDB or in JMX after invoking the **getLDAPSettings** method, by entering a value in the **Priority** field. If two or more LDAP Severs have the same priority, the order in which they are presented is alphabetic.
>
> ○ If you want to view the group information of LDAP users, set the **displayUsersGroup** parameter to **True**, and then invoke the **getLDAPGroupUsersChunk()** method. For details, see the "Search LDAP Users" section in the *HPE Universal CMDB Administration Guide*.

5. To change the values of LDAP authentication settings, locate the **configureLDAPServer** method. Enter the value for the **ldapURL** and the values for the relevant settings and click **Invoke**. The JMX MBEAN Operation Result page indicates whether the LDAP authentication settings were updated successfully.

> **Note:**
>
> ○ If you do not enter a value for a setting, the setting retains its current value.
>
> ○ If you want to delete an existing value, you have to delete the LDAP server and then reconfigure it by performing the steps described in this procedure.

6. After configuring the LDAP settings, you can verify the LDAP user credentials:

   a. Locate the **verifyLDAPCredentials** method.

   b. Enter the ldapHost, username, and password.

   c. Click **Invoke**.

   The JMX MBEAN Operation Result page indicates whether the user passes LDAP authentication.

> **Note:**

- A new setting **deleteLdapServer** is available, which allows you to delete an LDAP Server.

- The following methods are now global and have impact on all the UCMDB servers and LDAP servers: **allowLdapAuthentication**, **allowLdapSynchronization**, and **forceCaseMatchAuthentication**.

- All the other methods were updated and now have a new field named **ldapHost**.

7. Map LDAP user groups to UCMDB user groups.

   a. Launch your Web browser and enter the following address: **https://<server_ name>:8443/ucmdb-ui**, where **<server_name>** is the name of the machine on which HP Universal CMDB is installed.

   b. Enter your login parameters, and select an LDAP server from the drop-down list of LDAP servers.

      Note: The drop-down list will be visible only if you configured more than one LDAP server.

   c. Go to **Security > LDAP Mapping**, and from the drop-down list, select the LDAP Sever for which you want to map the user groups. For details, see the section about LDAP Mapping in the *HPE Universal CMDB Administration Guide*.

      Note:

      - Only the global settings will be visible in **Administration > Infrastructure Settings Manager > LDAP General** category.

      - Also, the following two settings are available in **Administration > Infrastructure Settings Manager > LDAP General** category: **Check subgroup existence in the LDAP Mapping** and **Enable LDAP Authentication in non interactive flows**.

8. **Important:** If you are configuring LDAP on a high availability environment, you must restart the cluster for the changes to take effect.

Note:

- Every LDAP user has a first name, last name, and email address saved in the local repository. If the value of any of these parameters that is stored on the LDAP server differs from the value in the local repository, the LDAP server values will overwrite the local values at each login.

- The value of the **userUID** setting must be unique across all LDAP servers.

How to Search LDAP Users

The JMX console provides a **getLDAPGroupUsersChunk()** method that allows you to search LDAP users. The method returns matched users in one or multiple chunks, and each chunk contains 100 users at most.

> **Note:** To search LDAP users, the LDAP server must support Virtual List View (VLV) and Server Side Sorting.

Follow these steps to search LDAP users from the JMX console:

1. Launch your web browser and enter the following address: **https://<server_name>:8443/jmx-console**, where **<server_name>** is the name of the machine on which HPE Universal CMDB is installed.

   You may need to log in with a user name and password.

2. Under **UCMDB**, click **UCMDB:service=LDAP Services** to open the Operations page.

3. Locate **getLDAPGroupUsersChunk**, and then specify the following parameters.

Parameter	Value
ldapHost	Enter the host name of the LDAP server. This parameter is mandatory.
searchInGroup	Enter the group's name if you want to search in a specific group. Default: search all groups.
searchByField	Specify one of the following fields: **uid**, **displayname**. The method then searches in the specified field. Default:**uid**.
searchValue	Enter the search value.
sortByField	Specify one of the following fields: **uid**, **displayname**. The method then sorts the search result by the specified field. Default:**uid**.
sortOrder	Specify the sort order of the search result. Default: `True` (in ascending order).
requestedChunkNumber	Specify which chunk of result that the method returns.

Parameter	Value
	Default: the first chunk.
multipleChunksRequest	Specify how many chunks of result that the method returns. The method returns the specified number of chunks starting from the first chunk. This parameter works only when **requestedChunkNumber** is not specified. Otherwise, the method only returns the chunk as specified in **requestedChunkNumber**.

> **Note:** If only **ldapHost** is specified, the method returns the first chunk of all users in all groups in the ascending order of the **uid** field.

4. Click **Invoke**.

> **Note:** The method can return the group information of LDAP users. To enable this functionality, you need to configure the LDAP server by using the **configureLdapServer()** method and set the **displayUsersGroup** parameter to `True`.

How to Configure the HP Universal CMDB Server with Confidential Manager

When working with HPE Universal CMDB, you should configure the secret and crypto-properties of the encryption, using the following JMX methods:

1. On the HPE Universal CMDB Server machine, launch the Web browser and enter the Server address, as follows: **https://<UCMDB Server Host Name or IP>:8443/jmx-console.**

   You may have to log in with a user name and password.

2. Under UCMDB, click **UCMDB:service=Security Services** to open the Operations page.

3. To retrieve the current configuration, locate the **CMGetConfiguration** operation.

   Click **Invoke** to display the Confidential Manager server configuration XML file.

4. To make changes to the configuration, copy the XML that you invoked in the previous step to a text editor.

   Locate the **CMSetConfiguration** operation. Copy the updated configuration into the **Value** box and click **Invoke.** The new configuration is written to the UCMDB Server.

5. To add users to Confidential Manager for authorization and replication, locate the **CMAddUser**

operation. This process is also useful in the replication process. In replication, the server slave should communicate with the server master, using a privileged user.

- **username**. The user name.

- **customer.** The default is ALL_CUSTOMERS.

- **resource.** The resource name. The default is ROOT_FOLDER.

- **permission.** Choose between ALL_PERMISSIONS, CREATE, READ, UPDATE, and DELETE. The default is ALL_PERMISSIONS.

Click **Invoke**.

6. If necessary, restart HP Universal CMDB.

In most cases there is no need to restart the Server. You may need to restart the Server when changing one of the following resources:

- Storage type

- Database table name or column names

- The creator of the database connection

- The connection properties to the database (that is, URL, user, password, driver class name)

- Database type

Note:

- It is important that the UCMDB Server and its clients have the same transport crypto-properties. If these properties are changed on the UCMDB Server, you must change them on all clients. (This is not relevant for the Data Flow Probe because it runs on the same process as the UCMDB Server—that is, there is no need for the Transport crypto-configuration.)

- Confidential Manager Replication is not configured by default, and can be configured if needed.

- If Confidential Manager Replication is enabled, and the Transportation **initString** or any other crypto-property of the master changes, all slaves must adopt the changes.

How to Set the IIS server as the Front-End Server for UCMDB

1. Launch the Web browser and enter the following address:

   http://<UCMDB server name>:<port>/jmx-console.

2. Click **UCMDB-UI:name=UI Server frontend settings** to open the JMX MBEAN View page.

3. Click the **setUseFrontendURLBySettings** method and enter the address of the IIS server as the value (**http://<IIS server name>:<port>**).

4. Click **Invoke**.

> **Note:** You cannot open the JMX Console from IIS. That is, basic authentication cannot be passed from Jetty.

How to Enable Secure Login for the JMX Console

To enable secure login for the JMX console,

1. Access the UCMDB JMX console: Launch a Web browser and enter the following address: **https://localhost:8443/jmx-console**. You may have to log in with a user name and password.

2. Locate **UCMDB:service=Ports Management Services** and click the link to open the Operations page.

3. Locate the **mapComponentToConnectors** operation.

4. To enable secure login for the JMX console, invoke the **mapComponentToConnectors** method with the following parameters:

   - **componentName**: jmx-console

   - **isHTTPS**: true

   - All other flags: false

5. Restart the server.

6. Log in to the JMX console using **https://** and port **8443** (default) or the one for https if it was changed.

   For example, **https://mymachine:8443**.

> **Note:** HP also recommends you setting a strong password for the **sysadmin** user and any other user that can access the JMX console.

How to Mark Sensitive Settings and Enable Storing Encrypted Data in the Database Using JMX

UCMDB administrators can mark sensitive settings and enabling storing encrypted values for the sensitive settings in the database by using the following JMX methods added in the **UCMDB:service=Settings Services** category:

- **listSensitiveSettings** - Returns the list of settings that are marked as sensitive.

- **markSettingAsSensitive** - Marks a setting as sensitive. Usually sensitive settings contain confidential data. If a setting is marked as sensitive, its data will be encrypted when stored in the database.

  > **Note:** A setting can be marked as sensitive only when its value has been changed. If a setting does not have a value or if the value is out of the box, then the setting cannot be marked as sensitive.

- **markSettingAsNonsensitive** - Marks a setting as non-sensitive. Non-sensitive settings will have the value stored in plain text in database. This method is also used to decrypt the sensitive settings you encrypted using the **markSettingAsSensitive** method.

To mark a setting as sensitive,

1. On the UCMDB server, launch the Web browser and enter the following address: **https://localhost:8443/jmx-console.**
2. Click **UCMDB:service=Settings Services** to open the JMX MBEAN View page.
3. Click the **markSettingAsSensitive** method.
4. Enter the name of the setting you would like to mark as sensitive.
5. Click **Invoke**.

To mark a setting as non-sensitive,

1. On the UCMDB server, launch the Web browser and enter the following address: **https://localhost:8443/jmx-console.**

2. Click **UCMDB:service=Settings Services** to open the JMX MBEAN View page.

3. Click the **markSettingAsNonsensitive** method.

4. Enter the name of the setting you would like to mark as non-sensitive.

5. Click **Invoke**.

To view a list of sensitive settings,

1. On the UCMDB server, launch the Web browser and enter the following address: **https://localhost:8443/jmx-console.**

2. Click **UCMDB:service=Settings Services** to open the JMX MBEAN View page.

3. Click the **listSensitiveSettings** method.

4. Click **Invoke**.

    A list of settings that are marked as sensitive is returned.

Note: The following existing settings are already encrypted in the database and cannot be marked as sensitive:

- **ha.cluster.authentication.keystore.password**

- **ha.cluster.authentication.shared.secret**

- **ha.cluster.message.encryption.keystore.password**

- **ssl.server.keystore.password**

- **ssl.server.truststore.password**

Starting from version 10.21, two new OOTB settings are marked as sensitive by default:

- **java.naming.ldap.search.password**

- **jetty.connections.http.probe.basicAuthentication.defaultPassword**

Starting from version 10.30, the following OOTB settings are encrypted by the master key all the time. They cannot be marked as non-sensitive, and will not display if you invoke the **listSensitiveSettings** JMX method:

- **java.naming.ldap.search.password**

- **java.naming.provider.url**

How to Set Shared Key for Encrypting or Decrypting the InfrastructureSettings.xml File Using JMX

UCMDB administrators can set a shared key for encrypting or decrypting the **InfrastructureSettings.xml** file on the UCMDB Server side or the Data Flow Probe/Integration Service side by using the **setSharedKey** JMX method.

Once you have set a shared key on the server side, make sure you set the same shared key on the Data Flow Probe/Integration Service side as well. This ensures that the Data Flow Probe/Integration Service can properly decrypt the **InfrastructureSettings.xml** file.

To set a shared key on the UCMDB Server side,

1. On the UCMDB server, launch the Web browser and enter the following address: **http://localhost:8443/jmx-console.**

2. Click **UCMDB:service=Discovery Manager** to open the JMX MBEAN View page.

3. Click the **setSharedKey** method.

4. Enter a new value in the **Value** field for the shared key.

5. Click **Invoke**.

To set a shared key on the Data Flow Probe/Integration Service side,

1. Access the Data Flow Probe/Integration Service JMX console: Launch a Web browser and enter the following address: **http://<Probe or integration service machine name or IP address>:1977**. If you are running the Data Flow Probe/Integration Service locally, enter **http://localhost:1977**.

   You may have to log in with a user name and password.

2. Locate the **Probe_<Probe Name> type=MainProbe** service and click the link to open the JMX MBEAN View page.

3. Click the **setSharedKey** method.

4. In the **Value** field, enter the same value you provided on the UCMDB Server side for the shared key.

5. Click **Invoke**.

> **Note:** If the Data Flow Probe is running in separate mode, make sure you set the shared key on both probeManager and probeGateway.

How to Configure CAC (Smart Card / PKI Authentication) Support for the Embedded UCMDB Browser

This section describes how to configure Smart Card Authentication or PKI Authentication (CAC) support for the embedded UCMDB Browser.

> **Note:** CAC support is only available when using Internet Explorer 10 or later.

1. Configure UCMDB to use LW-SSO authentication.

    For details on LW-SSO authentication, see"Enabling Login to Universal CMDB with LW-SSO" in the *HPE Universal CMDB Hardening Guide*.

2. Import the root CA and any intermediate certificates into the UCMDB Server Truststore as follows:

    a. On the UCMDB machine, copy the certificate files to the following directory on UCMDB:

    C:\HP\UCMDB\UCMDBServer\conf\security

    > **Note:** If your certificate is in Microsoft p7b format, you may need to convert it to PEM format.

    b. For each certificate, run the following command:

    ```
    C:\HP\UCMDB\UCMDBServer\bin\jre\bin\keytool.exe -import -v -keystore
    C:\HP\UCMDB\UCMDBServer\conf\security\server.truststore -file
    <certificate> - alias <certificate alias>
    ```

    c. Enter the UCMDB Server Truststore password (default **hppass**).

    d. When asked, **Trust this certificate?**, press **y** and then **Enter**.

    e. Make sure the output **Certificate** was added to the keystore.

3. Open the JMX console by launching the Web browser and entering the Server address, as follows:
    https://<UCMDB Server Host Name or IP>:8443/jmx-console.

You may have to log in with a user name and password.

4. Under UCMDB, click **UCMDB:service=Ports Management Services** to open the Operations page.

5. Click **mapComponentToConnectors**. In the mapComponentToConnectors service, do the following:

   ○ Map **ucmdb-ui**

     • Set **componentName** to **ucmdb-ui**.

     • Set only **isHTTPSWithClientAuth** to **true**, and click **Invoke**.

     • Click **Back to MBean**.

   ○ Map **ucmdb-browser**

     • Set **componentName** to **ucmdb-browser**

     • Set only **isHTTPSWithClientAuth** to **true**, and click **Invoke**.

     • Click **Back to MBean**

   ○ Map **root**

     • Set **componentName** to **root**.

     • Set **isHTTPSWithClientAuth** and **isHTTP** to **true**, and click **Invoke**.

6. Under UCMDB, click **UCMDB:service=Security Services** to open the Operations page. In the **loginWithCAC** service, set **loginWithCAC** to **true**, and click **Invoke**.

   You should now be able to log into UCMDB with **https://<UCMDB Server Host Name or IP>.<domainname>:8444**.

7. Assign roles or rights for each UCMDB Browser user in the UCMDB Server, as they will be created without roles or rights.

8. Restart the UCMDB Server.

Chapter 8: Installation and Migration Methods

This chapter includes:

How to Migrate DDMI Server Configuration Data to Universal Discovery

The following task describes how to migrate DDMI server configuration data to Universal Discovery. Migration tools, including Perl scripts and a JMX console are provided which automatically export DDMI server data and automatically import the data to UCMDB. In most cases, server data from DDMI is migrated into newly-created activities in UCMDB. For more information about activities in UCMDB, see the *HPE Universal CMDB Discovery and Integrations Content Guide*.

> **Note:**
>
> - Perform this task for each DDMI server that you want to migrate.
>
> - DDMI Aggregator Server configuration data is not supported.
>
> - Data Flow Probes that are members of probe clusters are not supported and should not be migrated.
>
> - To ensure Agent-related information, such as software utilization and callhome configuration data, is migrated, select the **Deploy** option in the Agent Deployment Actions field on the Agent Profile page.

This task includes the following steps:

1. "Prerequisite" on the next page

2. "Run the export script" on the next page

3. "Copy the archive file" on page 155

4. "Import the migration data" on page 155

5. "Results" on page 156

1. Prerequisite

    ○ Ensure that UCMDB is running.

        > **Note:** For information about installing UCMDB, see the interactive *HPE Universal CMDB Deployment Guide*.

    ○ Ensure that the DDMI server database is running.

    ○ (Optional) Back up the UCMDB database. For more information, see the documentation for your database product

    ○ If you want the discovery schedules that are contained in DDMI network profiles to migrate to Universal Discovery, ensure that the Force ARP Table to Be Read option is selected.

    ○ (Optional) If you do not know the **customer id** parameter for the customer you are migrating, do the following:

        i. In UCMDB, go to **Data Flow Management** > **Data Flow Probe Setup**.

        ii. In the **Domains and Probes** pane, select a Data Flow Probe and note the customer name at the top right of the window.

        iii. Go to the **JMX console** > **Customer & States** > **Show all Customers** method and note the **customer Id** that maps to the customer name.

    ○ (Optional) To validate that devices were migrated, run a Perl script on each DDMI Server that generates a device inventory report. The data in this report can be compared with data in UCMDB after migration, and it is useful for troubleshooting purposes. For more information, see How to Run the Device Inventory Report in the *HP Universal CMDB DDMI to Universal Discovery Migration Guide*

2. Run the export script

    a. Locate the **DDMIMigration.pl** script on the UCMDB Server at the following location:

        - **Windows: C:\hp\UCMDB\UCMDBServer\tools\migration**

        - **Linux: C:/opt/hp/UCMDB/UCMDBServer\tools\migration**

    b. Copy the script to any directory on each DDMI server that you want to migrate.

    c. For each DDMI server, open a Command prompt and navigate to the directory where you copied the script. At the Command prompt, run the following command:

        perl DDMIMigration.pl

You should see the following message:

"The migration data is successfully saved to **DDMIMigrationData.zip**".

Note:

- By default, the data is archived in a file called **DDMIMigrationData.zip**.

- Maximum amount of device groups that can be imported is 20. If device groups exceed 20, remove some groups and run the script again. Then, create the remaining management zones in Universal Discovery manually.

3. Copy the archive file

Copy the archive file that was created in step 2 to the following location on the UCMDB Server:

- Windows: **C:\hp\UCMDB\UCMDBServer\conf\discovery\customer_<customer id>**

- Linux: **C:/opt/hp/UCMDB/UCMDBServer\conf\discovery\customer_<customer id>**

where **customer id** is the value for the **customer id** parameter.

Note: Usually, this value is **1** by default.

4. Import the migration data

a. Open the JMX Console and go to **Discovery Manager** > **ImportMigrationDatafromDDMI**.

b. In the **importMigrationDataFromDDMI** method, the following parameters are displayed:

- **customerId.** The customer ID that you want to migrate.

- **isCreateActivity.**
  - **True.** Creates new activities in Management Zones. These activities contain the migrated data.
  - **False.** No activities are created. However, Management Zones are created.

- **Primary|Secondary Call Home Address.**The primary and the secondary call home IP addresses for the Data Flow Probe.

For example:

<UD_CallHomeIPAddressPrimary> , **<UD_CallHomeIPAddressSecondary>**

Note:

- If this field is left blank, the IP address of the Data Flow Probe is used.

- In some cases, data that is entered in these fields may not appear in the UCMDB Infrastructure activity. In these cases, reenter the data in the activity.

- The DDMI call home IP addresses are pre-populated, so it is not necessary to enter this information.

- **probeName.** The name of the Data Flow Probe to which to map the data.

- **configurationzipPackageName.** The name of the archive file that was created in step 2.

- **overrideGlobalConfig.**

  - **True.** The XML Enricher global configuration file in UCMDB is overwritten by the DDMI configuration file.

  - **False.** The XML Enricher global configuration file in UCMDB is not overwritten and the DDMI configuration file is ignored.

- **stopWhenConflict.**

  Specifies how to handle IP address range conflicts.

  - **True.** If overlapping IP address ranges exist in DDMI and UCMDB, no IP address ranges are imported to UCMDB.

  - **False.** If the same IP address range exists in UCMDB, only IP address ranges that are not in conflict are imported. Ranges that are in conflict are ignored. Additionally, Management Zones that contained the conflicted ranges are not imported.

5. Results

  ○ Success messages and warning messages are displayed.

  ○ In addition to the data that is contained in the archive file that was created in step 2, the following information is imported into UCMDB:

    - **Deployment credentials.** Credentials are imported and keys are regenerated automatically.

    - **SNMP configuration profile.**

    - **Device groups.**

    - **System configuration.**

    - **VMWare configuration.**

    - **XML Enricher configuration file.**

- **Certificates**
    - acstrust.cert
    - agentca.pem
    - acskeystore.bin
- **IP address ranges.**

○ Additionally, the following resources are imported:

- Pre-scan and post-scan scripts
- Scanner configuration files (.cxz)
- User SAI files

Send documentation feedback

If you have comments about this document, you can contact the documentation team by email. If an email client is configured on this system, click the link above and an email window opens with the following information in the subject line:

Feedback on JMX Reference Guide (Universal CMDB 10.30)

Just add your feedback to the email and click send.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to cms-doc@hpe.com.

We appreciate your feedback!