API Reference for CiscoWorks Network Compliance Manager 1.6

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

Text Part Number: OL-23304-01

API Reference for CiscoWorks Network Compliance Manager 1.6

Table of Contents

Getting Started

CiscoWorks Network Compliance Manager (NCM) is a powerful software solution for network configuration control with sophisticated Web and command-line interfaces for interactive use with NCM. The Java, Perl, and SOAP APIs add another dimension to NCM by integrating NCM with other software. You can link NCM to a variety of third-party and custom-built applications, such as ticketing, asset tracking, workflow, change request, and network management software solutions.

This document is intended for network engineering professionals who:

- Write scripts to automate device configuration.
- Are comfortable with basic Java and Perl programming, and have an understanding of database schema and access methods.
- Have knowledge of the NCM's Command Line Interface (CLI).
- Integrate various third-party systems with NCM, such as network management, workflow, and trouble ticketing solutions.

The document is divided into three chapters and two appendixes:

- Chapter 1: Java API
- Chapter 2: Perl API
- Chapter 3: SOAP API
- Appendix A: CLI/API Command Reference
- Appendix B: Installing the Perl API

Note: For NCM 1.6, there is an enhanced Perl API that uses standard Web services (SOAP) for simultaneously accessing NCM and functions with multiple NCM versions. In addition, the enhanced Perl API enables your scripts to simultaneously make connections to multiple NCM servers.

Chapter 1: Java API

Java is a modern, object-oriented language that can run on a variety of platforms. It lends itself to high performance, scalable, highly-available solutions. Java applications are also flexible and easily maintainable. Java is your best choice when using professional development resources, require high performance, and your solution is expected to be in used long term.

Requirements

To use the Java API, you must be running NCM 1.3.1 or higher. The server must be running and accessible to the client where your application runs via port 1099 (Java API). You must also have a valid license on the NCM server to use the Java API. A copy of the NCM client package is required to write the programs and run the examples.

Note: The server can be bound to a port other than 1099, but in this case, the session API must be explicitly provided with the port number.

Operating Systems

The Java API has been tested with the following operating systems:

Windows 2000 Professional with Service Pack 2

Windows 2000 Server with Service Pack 2

Windows 2003 Server

Red Hat Linux Enterprise AS (update 2 and 3)

Solaris 10

Java JDK

Download the Java JDK from Sun Microsystems at *http://java.sun.com/downloads/*

The Java API is tested with Java version 1.4.2 and Java 1.6.

NCM JAR Files

You will need two JAR files from the NCM server or client installation:

tuecontrol-client.jar

bcprov-jdk16-141.jar

See the instructions below for details on the location of these files.

Programming Model

The Java API is designed to expose a straightforward programming model with a relatively small set of objects to learn.

Centralized or Distributed Applications

The Java API can be accessed from your NCM server machine. This is the simplest programming model.

The Java API also enables you to run applications remotely from the NCM server. This means you can run NCM on machine A and API-based applications on machine B. In distributed software terminology, this is called remoting. By remoting your application, you will create a client/server application where NCM is the server and your application is the client. This might be desirable for load balancing, ease of setting up a development environment, security, or a variety of other reasons.

If you use remoting, you will need to configure your network to allow traffic on port 1099 (Java RMI) to reach the NCM server.

Request/Response

The Java API generally follows a request/response model. Your application makes a request via the `exec` method and waits for a response from the server. Details are explained in the “Programming with the Java API” section.

Threading model

The Java API is synchronous on the client side and asynchronous on the server side. This means that when your application makes a request, the calling thread in your application is blocked until a response is available from the server.

This response may mean that your command has been executed and the results returned (such as `list user` command, which immediately returns a list of users), or it may mean that an NCM task has been created and queued for future execution (such as the `get snapshot` command, which will schedule a NCM snapshot task).

If you want to issue multiple overlapped commands, you will need to use standard Java multi-threading techniques in your application.

Relationship to JDBC

You may notice a strong similarity between the Java API and certain JDBC calls. In particular, NCM returns results in a `ResultSet` object derived from JDBC. This makes it easier for developers familiar with JDBC to get up and running with the Java API. Some of the `ResultSet` methods are not applicable to NCM, and will return exceptions if used. These are detailed in the “Programming with the Java API” section.

Windows Installation

Installing from CD

This section describes how to install the Java API components from the installation CD.

From the NCM installation CD, use either the Server or Client installation options. With either of these options, you will get a copy of:

The Java Runtime Environment

The NCM client JAR

Any library JAR files necessary to run the NCM SDK

If you are running the SDK application on a machine that already has a NCM server installed, no further installation is required. Use the Client install if you will be connecting to a remote NCM server for your SDK application.

Note: The file paths referenced in this document assume you installed NCM to the default file location, `C:\NCM`. If you installed NCM to a different location, then replace `C:\NCM` with the root directory that you provided at installation.

Setup Java SDK

Install the Java SDK from Sun Microsystems or use the Java JRE that is installed with NCM. You need two JAR files to use the Java API:

<installation directory>/client/truecontrol-client.jar

<installation directory>/jre/lib/ext/bcprov-jdk14-119.jar

If you are using the Java JRE installed with NCM, the bcprov-jdk14-119 jar file is in the Java library path.

If you are using the Java SDK from Sun Microsystems, you must explicitly put the bcprov-jdk14-119jar file in your Java CLASSPATH as described below.

Java API JAR

The Java API JAR is located at `C:\NCM\client\truecontrol-client.jar`. However, it may be copied to another location.

Configuration Files

NCM and the Java API use several configuration files with an RCX extension. If you use the Java API on the same machine where you installed NCM, the RCX files will already be where they need to be (`C:\NCM\jre`). If you want to use the API on another machine, you need to manually copy the RCX files to the JRE/JDK directory on the other machine.

The RCX configuration files used by the Java API are:

messages.rcx
logging.rcx
commandlineclient.rcx

The Java API samples are located in `C:\NCM\client\sdk\examples\java`.

Java Documentation

The Java API documentation consists of the Javadocs for the API. Javadocs are located in `<install directory>\client\sdk\docs\api.`

Setting Up a Command-line Environment

If you are invoking javac and java from the command line, you can easily set up a command line environment to prepare to use the Java API. Append `truecontrol-client.jar` to your classpath.

Do not put the `truecontrol-client.jar` in jre/lib/ext. The NCM Java processes will not start.

Example: (Note that there should be no new line. The example appears to wrap due to documentation margin restrictions.)

```
set CLASSPATH=%CLASSPATH%;<installation directory>/truecontrol-
client.jar;<installation directory>/jre/lib/jar bcprov-jdk16-141.jar
```

To verify that your environment is correct, compile and run `Example0.java`. If `Example0.java` is not in your local directory, you can copy the file from `$INSTALL_DIRECTORY/client/sdk/examples/java/Example0.java` or you can “cd” into that directory.

Keep in mind that if you “cd” into the directory, you must have directory permissions to compile the class. Here is what you should see:

```
bash# INSTALL_DIRECTORY=/<install directory>
bash# export INSTALL_DIRECTORY
bash# export CLASSPATH=$CLASSPATH:.:$INSTALL_DIRECTORY/client/truecontrol-
client.jar:$INSTALL_DIRECTORY/jre/lib/ext/bcprov-jdk16-141.jar
bash# <install directory>/jre/bin/javac -d . Example0.java
bash# <install directory>/jre/bin/java com.rendition.api.examples.Example0

Starting Example0
Session connectivity verified

<install directory>/client/sdk/examples/java>
```

Unix Installation

Installing from CD

This section describes how to install the Java API components from the installer CD.

From the NCM installation CD, use either the Server or Client installation options. With either of these options, you will get a copy of:

- The Java Runtime Environment
- The NCM client JAR
- Any library JARs files necessary to run the NCM SDK

If you are running the SDK application on a machine that already has an NCM server installed, no further installation is required. Use the Client install if you will be connecting to a remote NCM server for your SDK application.

Note: The file paths referenced in this document assume you installed NCM to the default file location, `<install directory>/jre`. If you installed NCM to a different location, then replace `<install directory>/jre` with the root directory that you provided at install time.

Java API JAR

The Java API JAR is located at `<install directory>/jre/client/truecontol-client.jar`. Note that it might be copied to another location.

Configuration Files

NCM and the Java API use several configuration files with an RCX extension. If you use the Java API on the same machine that you installed NCM to, the RCX files will already be where they need to be, in the directory `<installed directory>/jre`. If you want to use the API on another machine, you need to manually copy the RCX files to the JRE/JDK directory on the other machine.

The RCX configuration files used by the Java API are:

messages.rcx
logging.rcx
commandlineclient.rcx

The Java API samples are located in:
`<installed directory>/client/sdk/examples/java`

Unix Installation Troubleshooting

If you see the following error:

```
com.rendition.api.RenditionAPIException: Could not connect to
server:10.101.22.21, LoginException:
javax.naming.CommunicationException [Root
exception is java.rmi.ConnectException: Connection refused to host:
127.0.0.1;
```

Where it says `Connection refused to host: 127.0.0.1`, even though you are connecting to NCM on a non-loopback address (10.101.22.21 in this error shown above), Java cannot determine the correct IP address for the host that is running the NCM Core.

To correct the issue, either:

Adjust the hostname resolution so that the hostname of the NCM Core host resolves to its correct IP address (typically this would be done by modifying the */etc/hosts* file)

or

Add: `wrapper.java.additional.5=-Djava.rmi.server.hostname=<correct-IP-address>` to `$NCM/server/ext/wrapper/conf/jboss_wrapper.conf`. and restart NCM.

Setting up an Integrated Development Environment

Setting up for an Integrated Development Environment (IDE) is similar to the command-line environment. You need to provide the location of `truecontrol-client.jar` to your IDE. In many editors, this is an option for the project. Details follow for selected IDEs.

Eclipse:

1. Go to File → New → Project.
2. Select Java Project and click Next.
3. Set the Project Name to Java API and click Next.
4. Click the Libraries tab and click Add External JARs.
5. Add truecontrol-client.jar.
6. Click Finish.

Jbuilder 5:

1. Go to the menu Project:Project Properties.
2. In the dialog box, select the Paths tab then the Required Libraries sub-tab.
3. Click Add then the New button.
4. Enter the name “NCM API.”

Navigate to the correct directory and select truecontrol-client.jar

Setting Up a Command-line Environment

If you are invoking javac and java from the command line, you can easily set up a command line environment to prepare to use the Java API. Append `truecontrol-client.jar` to your classpath.

Do not put the `truecontrol-client.jar` in jre/lib/ext. NCM’s Java processes will not start.

To verify that your environment is correct, compile and run `Example0.java`. If `Example0.java` is not in your local directory, you can copy the file from `$INSTALL_DIRECTORY/client/sdk/examples/java/Example0.java` or you can “cd” into that directory.

Keep in mind that if you “cd” into the directory, you must have directory permissions to compile the class. Here is what you should see:

```
bash# INSTALL_DIRECTORY=/<install directory>
bash# export INSTALL_DIRECTORY
bash# export CLASSPATH=$CLASSPATH:.:$INSTALL_DIRECTORY/client/truecontrol-
client.jar
bash# <install directory>/jre/bin/javac -d . Example0.java
bash# <install directory>/jre/bin/java com.rendition.api.examples.Example0

Starting Example0
Session connectivity verified

<install directory>/client/sdk/examples/java>
```

Programming with the Java API

If your Java environment is pointing to a different Java than the default NCM one, include the following jar to your CLASSPATH:

`$NCM/jre/lib/ext/bcprov-jdk14-119.jar`, where `$NCM` is your NCM local directory.

You will need this file to run the Java examples that are shipped with NCM.

Working with the Session object

All interaction with the Java API starts with a Session object.

Session contexts

`Session.open()` creates a session context for execution of commands. This method actually contacts the NCM server via Java RMI on port 1099, and authenticates the user using the supplied arguments. The server parameter is optional; if omitted, localhost will be contacted.

Make sure that you close the session context when done with it via the `Session.close()` method. Like file handles, there is a finite supply of sessions.

Session objects are thread-safe, so you may use the Session object across threads to do overlapping operations.

UserIDs and Permissions

When opening the session, you must provide a user name and password for a valid NCM user. NCM makes no distinction between the user identities used to log into the WebUI, CLI, or Telnet/SSH Proxy and those used to access the API.

Each NCM API call will be validated against the user identity provided to ensure the user has sufficient privileges to run the requested operation, just as the user's privileges would be validated by the WebUI, CLI, or Telnet/SSH Proxy.

It is recommended that you set up dedicated NCM users for API access, with appropriate privilege levels for the kinds of applications you are writing. For example, an application that only retrieves data from NCM might require a Limited Access user, whereas an application to remove out-of-date information from the system would require Admin privileges.

When calling `Session.open()`, note that the user name and password are case sensitive. If you provide bad authentication information, you will receive a NCMAPIException.

Executing Requests

You can send commands to the NCM server through the Session object.

Relationship between the API and the CLI or Telnet/SSH Proxy

`Session.exec()` is used to send a request to the NCM API. The commands accepted by `Session.exec()` are, with the exceptions noted below, syntactically identical to those accepted by the CLI or the Proxy interface interactive mode. You may find it convenient to test commands intended for your programs by telnetting to your server and manually entering the commands.

CLI-only Commands

All commands accepted by the CLI or Telnet/SSH Proxy are valid for `Session.exec()`, except for the `help`, `connect`, `os ping`, and `os traceroute` commands. The API does not support these.

Handling Results

This section covers the returned objects and exceptions thrown by `Session.exec()`.

Status

The return value from `Session.exec()` is a Result object. `Result.getSucceeded()` will return `true` if the command completed successfully; or `false` if the command failed. You can get extended information codes via `Result.getReturnStatus()`. The status codes vary based on the type of request; they are documented in the “Commands” section.

> Certain API commands schedule a task to be run on the server and then return immediately without waiting for the task to complete. Call `Result.getTex()` to get a string containing the Task ID number of the task that is created.
>
> Most of the commands accept an optional –sync argument to indicate that the API call should block until the task has completed and then return the task result. If –sync is used, `Result.getText()` returns the result output from running the task instead of the Task ID number. See Appendix A for information on which commands accept the –sync option.

Simple Results

If the command returns a simple String result, use Return.getString to examine the result. The commands with String results are shown in the “Commands” section.

Complex results: ResultSet type

Many commands return a complex result with many fields, or several rows of such field-based data. The commands with complex results are shown below in the Commands section.

The Java API uses JDBC’s ResultSet interface to provide access to complex results. You can learn more about this interface in numerous books and online resources for JDBC. The samples `Example1.java, Example2.java` and `Example3.java` show how to work with ResultSet data.

To interact with ResultSet data, you must know the valid columns and types for each command. This information is provided below in the Commands section, under the table heading Return Value(s). You can also use the metadata interface to work with ResultSets in a generic way, so that you do not have to hard code the data types being returned from a given command.

The `Result` object has a `toSring()` method that is useful for debugging to get more information about the results of your API calls.

Exceptions

The following exceptions are sent by Session.exec(). Details can be found in the javadocs.

NCMAPIException: Generic API exceptions.
ResultSetException: Thrown when incorrect method is used to retrieve a field from a ResultSet, e.g. calling getInt on a String field.
NotSupportedException: Thrown when an unsupported ResultSet method is called. See the javadocs for which methods are supported.

Metadata

Metadata (meaning data-about-data) describes the data fields returned in a ResultSet. You can use metadata to determine how many fields were returned in the result set, the name for each field, and the data type for each field. `ResultSet.getMetaData()` is the method that returns metadata for a result set.

`Example3.java` shows a useful application for metadata, processing any user-supplied command. You can see how metadata is required to print results from a command whose identity is not known at compile time.

Note: Developers familiar with C-based languages such as Java and C++ should note that the column indexes for all metadata methods are 1-based not 0-based.

Integration Hooks

Run External Application Tasks

NCM's Run External Application task enables you to invoke applications and scripts from within NCM. This includes the ability to run your own Java API applications. In other words, you can extend NCM's functionality by using this Java API to write your own application that integrates with outside applications and datasources.

Using the Web UI, you can configure NCM to invoke your own application when certain system events occur. Note that if you need to call out to third-party software from your custom application, you have several options:

Use that application's Java API, if one is provided.

Use that application's non-Java API via RMI.

Use a communication channel such as message queuing, CORBA, sockets, and so on.

Interact via the file system or databases.

Call that application directly via `Runtime.getRuntime().exec()`

Callbacks

There are two important callback methods from NCM to your Java code that you can use to customize the NCM engine:

The Approver interface

The Cleaner interface

Note that these callbacks cannot be remoted. The code must be present on the NCM server. You can provide a server-side stub which uses your own RMI calls to pass the call along to the client.

Also note that the following directions require you to modify NCM's configuration files. Make sure to keep a backup copy, as a corrupted configuration may make the server unstable.

Approver Callback Interface

The approver interface is provided to allow an external ticketing system to approve or deny a particular user's access to a device.

NCM will call the user-provided approver in the following circumstances:

Before the Telnet/SSH Proxy opens a device session – approveInterceptorSessionLogin() is invoked

Before a device configuration is modified – hasModifyConfigPermission() is invoked

Before a device group configuration is modified – hasGroupModifyConfigPermission() is invoked

Before any CLI command is processed – hasPermission() is invoked

See the javadoc comments for details on when these methods are invoked, and what parameters are passed. Note, some methods are overloaded.

Approver Use Cases

Here are two possible cases where this might be useful. The cases integrate NCM with a third-party ticketing system (3PT).

Case 1: External task approval

Network Engineer – Schedules a config deployment for ticket T and work request W.
NCM – Requests approval for change to device D with ticket T and work request W.
Ticketing System – Returns true or false with a reason R.
ONA – Lets the task run, or marks it as failed setting the Result to 'not approved by 3PT because R'.

Third-party ticketing system (3PT) should synchronously return true or false using internal data (such as time of day and ticket status) so no timeout is needed.

Case 2: External session approval

Network Engineer – Requests session on Device D for work request W.
NCM – Requests approval for connection to device D for work request W.
Ticketing System – Returns true or false with a reason R.
NCM – Starts the session or displays the error 'Session not approved by 3PT because R'.

Approver Coding

NCM will use the configuration file appserver.rcx to determine what class to use for the session approver. A default do-nothing (always approve) approver, `com.rendition.api.DefaultApprover`, is provided by NCM.

To install your own approver, follow these steps:

Code your own approver that implements the `com.rendition.api.Approver` interface

Modify "approver/className" option in appserver.rcx file, specifying your own class.

Build a JAR file that contains all your new classes and copy it into <installation directory>/server/ext/jboss/server/default/lib directory.

Cleaner Callback Interface

The cleaner interface returns custom actions upon user exiting a NCM device session. NCM will call the user-provided cleaner when the Telnet/SSH Proxy closes a device session.

Cleaner use case

Case 1: External change annotation

Network Engineer – Configures Device D for work request W. Closes session.

NCM – Calls cleaner for connection to device D for work request W.

Custom code – Calls out to ticketing system.

Ticketing System – Returns reason R for change.

Custom code – Calls Java API to copy reason R into custom data on device.

Cleaner Coding

NCM will use the configuration file appserver.rcx to determine what class to use for the session cleaner. A default do-nothing cleaner, `com.rendition.api.DefaultCleaner`, is provided by NCM.

To install your own cleaner, follow these steps:

Code your own cleaner that implements the `com.rendition.api.Cleaner` interface

modify "cleaner/className" option in appserver.rcx file, specifying your own class

build a JAR file that contains all your new classes and copy it into <installation directory>/server/ext/jboss/server/default/lib directory

User Permissions

The following table describes user permissions that are required to execute the CLI commands described in Appendix A. These roles are the default roles created by NCM. An administrator can create new permission groups and roles, and assign them to users.

User Permissions Matrix

			System Administration				
User	**Description**	**Reconfigure Devices** Log into enable mode View unmasked passwords Run configuration scripts Deploy configuration Change passwords	**Highly Sensitive** Manage users Delete historical information Edit/delete any users's tasks Define custom diagnostics	**Other** Administrative settings Authentication rules View all telnet/SSH sessions	**Group Tasks** Custom scripts & diagnostics Snapshots & polling Driver discovery Syslog configuration Password deployment Import FQDN lookup	**Modify NCM Information** Devices Groups Configuration comments	**View NA Information** Devices Groups Configurations Diagnostics Tasks System messages
Admin	Admins are highly trusted users responsible for administering the NCM application, managing users, setting policy, and running network-wide operations requiring a high degree of skill and care. They have permission to take any action in the NCM system on any device.	All Devices	**X**	**X**	**X**	**X**	**X**
Power User	Power users are highly trusted expert engineers allowed to perform most actions in the system. They can reconfigure and act on groups of devices in the system. They may be restricted to which devices they have permission to reconfigure.	Specified Devices		**X**	**X**	**X**	**X**

User Permissions Matrix

			System Administration				
User	**Description**	**Reconfigure Devices** Log into enable mode View unmasked passwords Run configuration scripts Deploy configuration Change passwords	**Highly Sensitive** Manage users Delete historical information Edit/delete any users's tasks Define custom diagnostics	**Other** Administrative settings Authentication rules View all telnet/SSH sessions	**Group Tasks** Custom scripts & diagnostics Snapshots & polling Driver discovery Syslog configuration Password deployment Import FQDN lookup	**Modify NCM Information** Devices Groups Configuration comments	**View NA Information** Devices Groups Configurations Diagnostics Tasks System messages
Full Access	Full Access users are qualified network engineers trusted with passwords to configure some or all devices in the network. They have permission to modify most information in the NCM database, and can reconfigure devices one-at-a-time but not in batch. They may be restricted as to which devices they have permission to reconfigure.	Specified Devices				**X**	**X**
Limited Access	Limited Access users are operator users that do not have passwords to configure network devices. They have permission to view but not modify most information in NCM. Sensitive information such as device passwords will be masked out. They cannot run batch operations or operations that reconfigure network devices.	**No** Devices					**X**

Chapter 2: Perl API

The Perl API enables NCM to communicate with external systems and vice-versa. The Perl API can be used to add and retrieve data to and from NCM.

Common tasks, such as adding devices into NCM and alerting third-party systems when a device configuration changes, can be programmatically accessed using the Perl API. Users who want to use other languages can automate their common functions using CLI or Telnet protocols.

Installing the Enhanced Perl API

Perl version 5.8 or later is required for installing on a Linux/Solaris platform. ActivePerl 5.8.x is required when installing on a Windows platform.

Note: If Perl is installed on your the NCM Server host, the NCM installer will automatically install the NCM Enhanced Perl API modules when NCM is installed. If you install Perl after installing NCM, install the NCM Enhanced Perl API with the Auto Installer.

If you want to install the Perl API after installing NCM, see Appendix B for instructions on installing the NCM Perl API.

Examples

There are Perl API examples in the demo directory. These examples show how to use the Perl API. Keep in mind that it is possible to run the examples without installing the Perl modules by remaining in the demo directory and supplying the relative (or full) path to each example, as in:

```
<installation directory>/client/perl_api/demo/list_users.pl –user
<NCM user> -pass <NCM password> -host <NCM server host>:80

<installation directory>/client/perl_api/demo/ list_users.pl –
user <NCM user> -pass <NCM password> -host <NCM server host>:80
```

If NCM is not running with the default HTTP port (80), replace 80 in the examples with the port number NCM is using.

Chapter 3: SOAP API

Using the WSDL file

As part of the installation, there are files named *api.wsdl.** in the *client/sdk/* directory of the core and client installations. These files represent the toolsets on which testing was performed. These toolsets can be stub compilers or other toolsets that can import WSDL files. The following information describes specific toolsets and known issues.

As with any WSDL file, the endpoint is specified in the <service> element. The value of the endpoint will be set to 'localhost' by default. This should be changed if it poses a problem for the target environment.

gSOAP - c/c++ stub compiler

For this toolset, use the *api.wsdl.gsoap* file.

When using the wsdl2h tool, there is a long list of warnings related to the use of a "type" versus an "element" in the types section. These warnings are benign. Normally, when using this tool, you use the generated .nsmap file. This structure contains URIs that do not work well with a server side implementation. As a workaround, use something similar to the following in the program:

```
SOAP_NMAC struct Namespace namespaces[] =
{
        {"SOAP-ENV", "http://schemas.xmlsoap.org/soap/envelope/", NULL,
NULL},
        {"SOAP-ENC", "http://www.w3.org/2003/05/soap-encoding",
"http://www.w3.org/*/soap-encoding", NULL},
        {"xsi", "http://www.w3.org/2001/XMLSchema-instance",
"http://www.w3.org/*/XMLSchema-instance", NULL},
        {"xsd", "http://www.w3.org/2001/XMLSchema",
"http://www.w3.org/*/XMLSchema", NULL},
        {"wsdl", "http://tempuri.org/wsdl.xsd", NULL, NULL},
        {"nas", "http://opsware.com/nas/", NULL, NULL},
        {NULL, NULL, NULL, NULL}
};
```

axis1.4 and wsdl2java - Apache axis stub compiler for java

For this toolset, use the *api.wsdl.axis* file.

Java has a hard limit of 256 parameters to a method. This tool generates constructors for each of the objects it finds in the WSDL file. One of these objects, Row, contains more than 256 members. As a result, one of the generated constructors has too many arguments. This can be solved in your make or ant environment by doing a deletion or replacement, for example:

```
sed -i -e '/public Row($$/, /}/d' $OUT_CLASS_DIR/Row.java
```

axis2 and wsdl2java - Apache axis2 stub compiler for java

For this toolset, use the *api.wsdl.axis2* file.

The constructor problem described in the previous section does not apply to this tool for client-side generated code.

The generated stub will be located at:

src/com/opsware/nas/_72/NetworkManagementApiStub.java

You must modify the generated stub to work properly against the service to avoid the runtime exception regarding an incorrect subelement *Result*.

wsdl2py - Python stub compiler from the ZSI package

For this toolset, use the *api.wsdl.wsdl2py* file.

Once the stubs are created, there is a change which must be made to the resulting files. One of the exported functions is called "import". Since this is a reserved word in Python, using import creates a conflict. The Python stub compiler creates a method for each of the exposed methods of the service. For this module to compile, you must rename this method. As part of the build process, doing a simple replacement in the stub file after it is generated solves the problem:

```
sed -i -e 's/def import/def _import/g' `grep -l "def import" * | grep -
v Makefile `
```

The reason for the embeded grep is because different versions of ZSI will generate different stub names.

SOAPpy - Python module

For this toolset, use the *api.wsdl.soappy* file.

This module allows the dynamic loading of a file, rather than creation of stubs like other tools. One way to make usage easier is making sure the endpoint in the <service> tag points to the desired NCM server before using the file to create the WSDL.Proxy object. By default, this file comes with an endpoint of localhost. It may be easier to modify the file that this tool uses rather than explicitly changing the endpoint when using this toolset.

.NET wsdl.exe compiler

For this toolset, use the *api.wsdl.dotnet* file.

There are compabibility issues with Visual Studio .NET 2003. At this time, only wsdl.exe from Visual Studio .NET 2005 is supported. If used with VS 2003 is required, generating stubs in VS 2005 and using 2003 should solve the issue.

wsdl.exe does not create custom exceptions based on the <fault> information in the WSDL. To handle exceptions thrown by the server, something like the following will be required in the program:

```
try
{
           result = api.login( inParms );
}
catch( System.Web.Services.Protocols.SoapException e )
{
    Console.WriteLine( "Caught exception: " + e.Message );
    //  e.Detail contains the parsable node
}
```

Appendix A: CLI/API Command Reference

Note that “Status” means there is no Text or ResultSet, only getReturnStatus() can be called when using the command.

Commands

activate device

Mark a device as activated.

Synopsis

activate device [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID

Return Type

Status

Examples

activate device -ip 192.0.2.10
activate device -ip "East Site:192.0.2.10"

add advanced script

Add a new advanced script.

Synopsis

add advanced script -name <Name> [-description <Description>] [-scripttype <Script Type>] [-family <Device Family>] -language <Script Language> [-parameters <Parameters>] -script <Script Text> [-sitename <Site name>]

Description

-name - Name for the new advanced script
-description - Description for the new advanced script
-scripttype - Script type (i.e. user defined subcategory)

-family - Device family for the new advanced script
-language - Language for the new advanced script - must be a supported language such as Expect or Perl
-parameters - Command line parameters for the new advanced script
-script - Script text
-sitename - Site name

Return Type

Status

Examples

add advanced script -name "Extended Ping" -description "Run extended ping to desired address" -scripttype "Troubleshooting scripts" -family "Cisco IOS" -language "Expect" -parameters "-l /usr/etc/log.txt" -script "send \"extended ping $Target_IP$\"" -sitename "Default Site"

add authentication

Add device password information.

Synopsis

add authentication -loc <Location> [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-snmpro <Read only community string(s)>] [-snmprw <Read write community string(s)>] [-snmpv3user <SNMPv3 Username>] [-snmpv3authpw <SNMPv3 Authentication Password>] [-snmpv3encryptpw <SNMPv3 Encryption Password>] [-user <Username>] [-passwd <Password>] [-enableuser <Enable username>] [-enablepasswd <Enable password>] [-connectionmethods <Connection methods>] [-accessvariables <Access variables>] [-start <Task start date>] [-appendsnmpro] [-appendsnmprw] [-sync] [-group <Group name>] [-site <SiteName>] [-rule <Rule name>] [-rulehostname <Rule hostname>] [-ruledevicegroup <Rule device group>] [-iprangestart <Rule ip start range>] [-iprangeend <Rule ip end range>]

Description

This command can modify passwords on a specific device or device group, or merely update what the system knows of a device's or network's password information. The -ip option provides information specific to the device. Otherwise, the command adds a network-wide password rule to the system. When using this command to modify passwords on a device, the modification operation is actually a scheduled task.

-loc - The location to which password information should be written. Valid values for this argument are "db", "device", and "group". "db" tells the command that password information should be changed only in the system's database. "device" tells the command that the password changes should be made on the device as

well and "group" performs the same function as "device" but across all devices in the group.
-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.: The device to which this password information should apply.
-host - A valid hostname: An existing device to which this password information should apply.
-fqdn - A valid Fully Qualified Domain Name: An existing device to which this password information should apply.
-deviceid - A device ID
-snmpro - When used in conjunction with -loc db, this argument is taken as a single community string understood by the system as THE read only community string for the device or network. When used in conjunction with -loc device, this argument is taken as a comma-seperated list of read only community strings to be, either set on the device, or appended to an existing list of read only community strings (depends on whether or not the -appendsnmpro flag was supplied.)
-snmprw - When used in conjunction with -loc db, this argument is taken as a single community string understood by the system as THE read write community string for the device or network. When used in conjunction with -loc device, this argument is taken as a comma-seperated list of read write community strings to be, either set on the device, or appended to an existing list of read write community strings (depends on whether or not the -appendsnmprw flag was supplied.)
-snmpv3user - When used in conjunction with -loc db, this argument is taken as the username for snmpv3 access.
-snmpv3authpw - When used in conjunction with -loc db, this argument is taken as the authentication password for snmpv3 access.
-snmpv3encryptpw - When used in conjunction with -loc db, this argument is taken as the encryption password for snmpv3 access.
-user - Username.
-passwd - Password.
-enableuser - ADDITIONAL username to get to "enable" mode.
-enablepasswd - ADDITIONAL password to get to "enable" mode.
-connectionmethods - The methods used by the system to connect to devices. Can be telnet, serial_direct, or SSH.
-accessvariables - To override variables in the script, such as prompts.
-start - YYYY:MM:DD:HH:mm. The first date on which the task will run. Use this option only if the argument to the -loc flag is "device".
-appendsnmpro - Supply this option if read only community strings should be appended to any existing on the device. Use this option only if the argument to the -loc flag is "device".
-appendsnmprw - Supply this option if read write community strings should be appended to any existing on the device. Use this option only if the argument to the -loc flag is "device".
-sync - Indicates that the command should return only after the password change task is complete. Do not use this option with -start.
-group - The group name for performing this command across all devices in a group.
-site - The site partition this rule will be applied to. Default to be global
-rule - the rule name to be added

-rulehostname - Hostname, the rule applies to
-ruledevicegroup - Device group name, the rule applies to
-iprangestart - IP start range, the rule applies to
-iprangeend - IP end range, the rule applies to

Return Type

String

Examples

add authentication -loc db -ip 192.0.2.10 -passwd fish -snmpro public -enablepasswd 31337
add authentication -loc db -ip 192.0.2.10 -passwd old -enablepasswd joshua -snmpro public -snmprw public
add authentication -loc device -ip 192.0.2.10 -passwd limited -enablepasswd full
add authentication -loc device -ip 192.0.2.10 -passwd some -enablepasswd all -snmprw brillig,slithy,toves,gire -appendsnmprw -sync
add authentication -loc device -ip 192.0.2.10 -passwd less -enablepasswd more -snmpro foo,bar,fork,snork -start 2004:02:29:23:59
add authentication -loc group -group MyDevices -passwd less -enablepasswd more -snmpro foo,bar,fork,snork -start 2004:02:29:23:59
add authentication -loc db -rule "rule 1" -rulehostname DALAB-C2600-NAT
add authentication -loc db -site DefaultSite -rule "rule 2" -ruledevicegroup DeviecGroup1
add authentication -loc db -rule "rule 3" -iprangestart 172.30.1.1 -iprangeend 172.30.1.5

add command script

Add a new command script.

Synopsis

add command script -name <Name> [-description <Description>] [-scripttype <Script Type>] -mode <Mode> [-driver <Driver List>] -script <Script Text> [-sitename <Site name>]

Description

-name - Name for the new command script
-description - Description for the new command script
-scripttype - Script type (i.e. user defined subcategory)
-mode - Command script mode
-driver - List of applicable drivers - provided as a comma separated list of internal driver names
-script - Script text
-sitename - Site name

Return Type

Status

Examples

add command script -name "Extended Ping" -description "Run extended ping to desired address" -scripttype "Troubleshooting scripts" -mode "Cisco IOS enable" -driver "CiscoIOSGeneric,CiscoIOSSwitch" -script "extended ping $Target_IP$" -sitename "Default Site"

add device

Add a device to the system.

Synopsis

add device -ip <IP address> [-hostname <Host name>] [-comment <Comment>] [-description <Device name>] [-model <Device model>] [-vendor <Device vendor>] [-domain <Domain name>] [-serial <Serial number>] [-asset <Asset tag>] [-location <Location>] [-unmanaged <Unmanaged>] [-nopoll <Do not poll>] [-consoleip <Console IP address, if using console server>] [-consoleport <Console Port>] [-tftpserverip <TFTP server IP address, if using NAT>] [-natip <NAT IP address>] [-useconsoleserver <true or false>] [-accessmethods <Comma-separated list of access methods>] [-hierarchylayer <Hierarchy layer>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device will be put in.
-hostname - The device's host name
-comment - Additional information regarding the device.
-description - The descriptive name of the device (informational only).
-model - The device's model (such as 2620).
-vendor - The device's vendor (such as Cisco).
-domain - A fully qualified domain name (such as www.google.com).
-serial - The device's serial number.
-asset - The device's asset tag.
-location - The device's location.
-unmanaged - 0: Mark this device as managed by the system. 1: Mark this device to be unmanaged by the system.
-nopoll - 0: Mark this device to be polled for changes. 1: Mark this device as not to be polled for changes.
-consoleip - a.b.c.d where 0 <= a,b,c,d <= 255
-consoleport - The port number
-tftpserverip - a.b.c.d where 0 <= a,b,c,d <= 255
-natip - a.b.c.d where 0 <= a,b,c,d <= 255
-useconsoleserver - true, if the device uses a console server. false, if the device does not. If this option is not provided, it is assumed that the device does not use a console server.

-accessmethods - A comma-separated list of access methods, or "none". The set of access methods: {telnet, ssh, rlogin, SCP, FTP, TFTP, SNMP, snmp_noauthnopriv, snmp_authnopriv, snmp_authpriv}. If this option is not provided, the system will try all access methods when attempting to connect to the device.
-hierarchylayer - This device attribute is used in diagramming. When you config a network diagram, you can select which hierarchy layers on which to filter. Valid values include: (core, distribution, access, edge and "layer not set").

Return Type

Status

Examples

add device -ip 192.0.2.10
add device -ip "East Site:192.0.2.10"
add device -ip 192.0.2.10 -model 3460 -vendor Cisco
add device -ip 192.0.2.10 -comment "the web server." -domain www.minibosses.com
add device -ip 192.0.2.10 -consoleip 192.0.2.10 -consoleport 62888 -useconsoleserver true -accessmethods ssh,SNMP

Add Device Context

Synopsis

add device context -deviceid <Device ID> -contextvariables <Context Variables>

Description

For drivers that support it, add a virtual context to a device

-deviceid - The device ID to add a context to

-contextvariables - A comma separated list of driver specific variables and values that are required to create a context on the given device. These variables can be found using the 'list device context variables' command.

Return Type

VO:ScheduleTaskVO with columns:

approvalPriority
approvalStatus
approvalUsers
approveByDate
comments
completeDate
coreID
createDate
csvData
csvGroupTask

deviceDataID
deviceGroupID
deviceID
duration
expensive
failedChildCount
failureType
lastModifiedDate
parentTaskID
pendingChildCount
priority
repeatCount
repeatEndDate
repeatInterval
repeatType
repeatWeekday
reservationEndDate
reservedDuration
reservedDurationType
result
resultConfigID
retryCount
retryInterval
runASAP
scheduleDate
scheduleTaskID
siteID
startDate
status
subTask
subTasks
succeededChildCount
systemTask
taskCustom1
taskCustom2
taskCustom3
taskCustom4
taskCustom5
taskCustom6
taskData
taskName
taskType
taskUserID
ticketNumber
userGeneratedChange

Examples

add device context -deviceid 749 -contextvariables "context_name=foo,config_location=disk:,config_filename=bar.cfg"

add device relationship

Add a new Device relationship between two devices.

Synopsis

add device relationship -origindeviceid <Originating Device ID> -destdeviceid <Destination Device ID> -relationshiptypeid <Relationship Type ID>

Description

Adds a new device relationship between the two given devices.

-origindeviceid - Specify the Originating Device ID for the relationship

-destdeviceid - Specify the Destination Device ID for the relationship

-relationshiptypeid - Specify Relationship type ID, 1: PC_USER , 2:PC_GENERIC , 3:PC_ADMIN_CONTEXT, 4:PEER_USER, 5:PEER_GENERIC

Return Type

VO:DeviceRelationshipVO with columns:

createdBy
destinationDeviceID
deviceRelationshipID
lastModifiedDate
originatingDeviceID
relationshipTypeID

Examples

add device relationship -origindeviceid 281 -destdeviceid 351 -relationshiptypeid 2

add device template

Add a new device template.

Synopsis

add device template -hostname <Device name> [-driver <Driver name>] [-comment <Comment>] [-description <Description>] [-model <Device model>] [-vendor <Device vendor>] [-location <Location>] [-customname <Customname>] [-customvalue <Customvalue>] [-accessmethods <Comma-separated list of access methods>] [-hierarchylayer <Hierarchy layer>] [-sitename <Site Name>]

Description

-hostname - A valid name

-driver - The driver name in short form

-comment - Additional information regarding the device template.

-description - The descriptive name of the device template (informational only).

-model - The device template's model (such as 2620).

-vendor - The device's vendor (such as Cisco).

-location - The device's location.

-customname - The custom field name

-customvalue - The custom field value

-accessmethods - A comma-separated list of access methods, or "none". The set of access methods: {telnet, ssh, rlogin, SCP, FTP, TFTP, SNMP,snmp_noauthnopriv, snmp_authnopriv_sha, snmp_authnopriv_md5, snmp_authpriv_sha_des, snmp_authpriv_sha_aes, snmp_authpriv_sha_aes192, snmp_authpriv_sha_aes256, snmp_authpriv_md5_des, snmp_authpriv_md5_aes, snmp_authpriv_md5_aes192, snmp_authpriv_md5_aes256}.

-hierarchylayer - This device attribute is used in diagramming. When you config a network diagram, you can select which hierarchy layers on which to filter. Valid values include: (core, distribution, access, edge and "layer not set").

-sitename - The Site name in which the template belongs to.

Return Type

STATUS

Examples

- add device template -hostname SampleTemplate -driver 3ComSuperStackIII -comment "Test Comment"
- add device template -hostname CISCOTemplate -customname Owner -custo

add device to group

Add a device to a device group.

Synopsis

add device to group [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] -group <Device group>

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-group - The name of the device group to which the device should be added.

Return Type

Status

Examples

add device to group -ip 192.0.2.10 -group tech-dev
add device to group -ip "Default Site:192.0.2.10" -group tech-dev

add diagnostic

Add a new custom diagnostic script.

Synopsis

add diagnostic -name <Name> [-description <Description>] -mode <Mode> [-driver <Driver List>] -script <Script Text> [-sitename <Site name>]

Description

-name - Name for the new diagnostic
-description - Description for the new diagnostic
-mode - Command script mode
-driver - List of applicable drivers - provided as a comma separated list of internal driver names
-script - Diagnostic script text
-sitename - Site name

Return Type

Status

Examples

add diagnostic -name "Show IP CEF" -description "Gather IP CEF information" -mode "Cisco IOS enable" -driver "CiscoIOSGeneric,CiscoIOSSwitch" -script "show ip cef" -sitename "Default Site"

add event

Add an event.

Synopsis

add event -message <Event> [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>]

Description

An email message (containing the event) will be the result of an added events if the system is configured to send email for added events.

-message - The text of the event
-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name

Return Type

Status

Examples

add event -ip 192.0.2.10 -message "Connectivity to the border router has been restored."
add event -message "This is a test of the emergency broadcast system."

add event rule

Add event rule.

Synopsis

add event rule -name <Event Rule Name> -action <Event Action> [-receiverhost <Hostname or IP Address>] [-receiverport <Port>] [-events <List of Event Types>] [-community <Community String>] [-eventtemplate <Event Template Filename>] [-eventtext <Event Text>] [-site <Site Name>]

Description

Add new event rule. It will subscribe provided host to the system events.

-name - The name identifier for event rule
-action - event type, for now only support snmp and integration, use -action snmp/integration
-receiverhost - A valid hostname or ip address
-receiverport - A numeric port, if not provided, then 162 will be used
-events - List of event types, separated by column. If not provided, then ALL will be used
-community - Community string, if not provided, then public will be used

-eventtemplate - Specify the absolute path to the file which contains the event text template. The file must be directly accessible by the system.
-eventtext - Specify the event text
-site - Name of teh site the rule will be added to. The rule will be global if site name is null or not specified

Return Type

Status

Examples

add event rule -name Name1 -receiverhost host1 -action snmp -community private -events "Device Added:Device Deleted"
add event rule -name Name2 -receiverhost host2 -action snmp -site SiteA
add event rule -name Name3 -action integration -events "Device Added" -site "Default Site"
add event rule -name Name4 -action syslog -receiverhost 10.2.3.4 -events "Device Added" -eventtext "Device $DeviceID$ added to Network Automation"

add group

Add a group to the system.

Synopsis

add group -name <Name> -type <Type> [-comment <Comment>] [-shared <Shared>]

Description

-name - The name of the group to add.
-type - The type of the group to add. "device" and "user" are the valid values for this option.
-comment - Additional information about the group.
-shared - 1 if the group is shared, 0 if it is not.

Return Type

Status

Examples

add group -name "border routers" -type device -comment "The group containing all border routers."

add group to parent group

Add a device group to a parent device group.

Synopsis

add group to parent group -parent <Parent group name> -child <Child group name>

Description

-parent - Name of the parent group
-child - Name of the child group

Return Type

Status

Examples

add group to parent group -parent "North America" -child "West Region"

add image

Add images to database.

Synopsis

add image [-site <site name>] -imageset <imageset name> -images <images> [-driver <driver name>] [-model <model name>] [-memory <minimum system memory (in bytes)>] [-processor <processor name>] [-bootrom <BootROM name>]

Description

Add images to database. Must specify either driver or model

-site - The site the image will be applicable to. The image will be global if site is not specified.
-imageset - The imageset the images will add to.
-images - The images to add. The paths specified by this option must point to files accessible by the management server. Files must be placed on the management server first.
-driver - The driver the images required.
-model - The device model the images required.
-memory - The minimum system memory required (in bytes) for images.
-processor - The hardware required for images.
-bootrom - The BootROM required for images.

Return Type

Status

Examples

add image -site SiteA -imageset fooset -images c:\\data\\bar.bin -driver CiscoPIX

add image -imageset fooset -images c:\\data\\bar.bin,c:\\images\\foobar.bin -model "WS-C2924M-XL-EN (C2900XL series)"
add image -imageset fooset -images /var/upload/bar.bin,/var/upload/foo.bin -driver CiscoPIX

add imageoption

Add information for device.

Synopsis

add imageoption [-imagemodels <device model names (separeted by ',')>] [-imageprocessors <device processor names (separeted by ','),>] [-imagebootroms <device bootROM names (separeted by ','),>]

Description

Add information for device which is not under management but needed for software image update.

-imagemodels - device model to be added.
-imageprocessors - device processors to be added.
-imagebootroms - device BootROMs to be added.

Return Type

Status

Examples

add imageoption -imagemodels model1,model2 -imageprocessors PP3,PP4

add ip

Add new secondary ip address.

Synopsis

add ip -ipvalue <Value> [-deviceip <Device IP address>] [-comment <Comment>] [-usetoaccess <Use to Access Device>] [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-ipvalue - The ip value a.b.c.d where 0 <= a,b,c,d <= 255
-deviceip - The device's ip address a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-comment - Additional information regarding the device.

-usetoaccess - Use this IP Value to access its device, 0 - yes, 1 - no, default - no
-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID

Return Type

Status

Examples

add ip -deviceip 192.0.2.10 -ipvalue 192.0.2.10 -comment "my own ip"
add ip -deviceip 192.0.2.10 -ipvalue 192.0.2.10 -usetoaccess 0
add ip -deviceid 1401 -ipvalue 192.0.2.10 -usetoaccess 0

add metadata

Add a piece of custom data to be associated with a specific field and associated entity.

Synopsis

add metadata -fieldid <Metadata Field ID> [-data <Data>] -associatedtableid <Matching Row ID>

Description

-fieldid - Field ID the data is to be associated with
-data - Data to be associated, if not included, data is null
-associatedtableid - ID of the associated row the data coresponds to

Return Type

MetadataVO with columns:

associatedTableID
data
fieldID
lastModifiedDate
metadataID

Examples

add metadata -fieldid 121 -associatedtableid 21031
add metadata -fieldid 121 -data Room101 -associatedtableid 21031

add metadata field

Define a custom data field for a specific table.

Synopsis

add metadata field -fieldname <Field Name> [-fieldvalues <Field Values>] [-inuse <In Use>] [-flags <Allow HTML>] -associatedtable <Associated Table>

Description

-fieldname - Name of the field to be added
-fieldvalues - List of comma separated values that the field is restricted to. If not specified, the value for this field is not restricted
-inuse - Turns the field on or off. 1 is on, 0 is off. When the field is off, it will not be displayed with the other custom fields.
-flags - Used for allowing HTML in the field value. 1 is allow, 0 is disallow. If disallowed, HTML will be escaped for displaying.
-associatedtable - The table to associate this field with

Return Type

MetadataFieldVO with columns:

associatedTable
fieldDataType
fieldID
fieldName
fieldValues
flags
inUse
lastModifiedDate

Examples

add metadata field -fieldname Room -fieldvalues 101,102,103,104 -inuse 1 -flags 0 -associatedtable RN_DEVICE
add metadata field -fieldname Building -inuse 1 -flags 0 -associatedtable RN_DEVICE

add parent group

Add a parent group to the system.

Synopsis

add parent group -name <Name> -type <Type> [-comment <Comment>]

Description

-name - The name of the parent group to add.
-type - The type of the parent group to add. "device" is currently the only valid argument to this option.
-comment - Additional information about the parent group.

Return Type

Status

Examples

add parent group -name "North America" -type device -comment "Parent group to roll up East, Central and West regions."

add partition

Add a partition to view.

Synopsis

add partition -viewname <Viewname> -name <Name> [-comment <Comment>]

Description

-viewname - The name of the view this partition goes to.
-name - The name of the partition to add.
-comment - Additional information about the partition.

Return Type

Status

Examples

add partition -viewname "Site" -name Redmond -comment "Redmond Site"

add service type

Add a custom service type to a device.

Synopsis

add service type [-ip <IP Address>] [-deviceid <Device ID>] -service <Service Name>

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-deviceid - device id
-service - service

Return Type

Status

Examples

add service type -deviceid 3508 -service Custom

add system message

Add a system message.

Synopsis

add system message -message <System Message> [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

An email message (containing the system message) will be the result of an added system message if the system is configured to send email for an added event.

-message - The text of the system message
-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID

Return Type

Status

Examples

add system message -ip 192.0.2.10 -message "Connectivity to the border router has been restored."
add system message -message "This is a test of the emergency broadcast system."

add user

Add a user to the system.

Synopsis

add user -u <Username> -p <Password> -fn <First name> -ln <Last name> [-email <Email address>] [-aaausername <Username>] [-aaapassword <AAA Password>] [-useaaaloginforproxy <Use AAA Logins for Proxy (yes|no)>] [-extauthfailover <Allow External Auth Failover (yes|no)>]

Description

-u - Username
-p - Password
-fn - First name
-ln - Last name
-email - Email address
-aaausername - AAA username for this user.
-aaapassword - AAA password for this user.
-useaaaloginforproxy - Whether to user AAA logins for the Proxy Interface for this user (yes|no).
-extauthfailover - Whether to allow external auth failover for this user (yes|no).

Return Type

Status

Examples

add user -u johnd -p fish -fn john -ln doe -email johnd@example.net

add user to group

Add a user to a user group.

Synopsis

add user to group -u <Username> -g <User group name>

Description

-u - Username
-g - User group name

Return Type

Status

Examples

add user to group -u johnd -g "User Group 1"

add vlan

Adds a Vlan - Add ports

Synopsis

add vlan [-deviceid <Device IP address>] [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] -vlanid <Vlan ID> [-vlanname <Vlan Name>] [-addports <Add Port IDs>] [-start <Task start date>] [-rep

<Task repeat period>] [-sync] [-sessionlog <true or false>] [-retrycount <Retry count>] [-retryinterval <Retry interval>] [-comment <Snapshop comment>] [-presnapshot <true or false>] [-postsnapshot <true, false or task>] [-priority <Task priority>]

Description

-deviceid - The device's ip address a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-host - A valid hostname

-fqdn - A valid Fully Qualified Domain Name

-deviceid - A device ID

-vlanid - Vlan ID to add

-vlanname - Name for Vlan added

-addports - Ports that need to be added to the Vlan

-start - YYYY:MM:DD:HH:mm. The first date on which the task will run.

-rep - (#min | #:# | #days | #weeks | #months) where # is a positive integer. #:# is hours:minutes--the two integers do not have to be the same. Do not use this option with -sync.

-sync - Indicates the command should return only after the snapshot retrieval task is complete. Do not use this option with -rep or -start.

-sessionlog - If true a complete session log will be saved with this task.

-retrycount - The number of times to retry the task if it fails.

-retryinterval - The number of seconds between retries.

-comment - An optional comment about the snapshot.

-presnapshot - If false, this indicates that the snapshot that runs before the script should be skipped.

-postsnapshot - If false, this indicates that the snapshot that runs after the script should be skipped. If "task", this indicates that snapshot after the script should run as a separate task.

-priority - Task priority value (1, 2, 3, 4 or 5). Invalid priority will be changed to an appropriate value automatically.

Return Type

String

Examples

add vlan -deviceid 2801 -vlanid 112 -addports 12,13,14

add vlan trunk

Adds a Vlan trunk - Add vlans to the trunk

Synopsis

add vlan trunk [-deviceip <Device IP address>] [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] -portname <Port Name> [-nativevlanid <Native vlan ID>] -addvlanids <Add Vlan IDs> [-start <Task start date>] [-rep <Task repeat period>] [-sync] [-sessionlog <true or false>] [-retrycount <Retry count>] [-retryinterval <Retry interval>] [-comment <Snapshop comment>] [-presnapshot <true or false>] [-postsnapshot <true, false or task>] [-priority <Task priority>]

Description

-deviceip - The device's ip address a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-host - A valid hostname

-fqdn - A valid Fully Qualified Domain Name

-deviceid - A device ID

-portname - trunk port name to add

-nativevlanid - specify a native or default vlan id

-addvlanids - vlan ids to add to trunk

-start - YYYY:MM:DD:HH:mm. The first date on which the task will run.

-rep - (#min | #:# | #days | #weeks | #months) where # is a positive integer. #:# is hours:minutes--the two integers do not have to be the same. Do not use this option with -sync.

-sync - Indicates the command should return only after the snapshot retrieval task is complete. Do not use this option with -rep or -start.

-sessionlog - If true a complete session log will be saved with this task.

-retrycount - The number of times to retry the task if it fails.

-retryinterval - The number of seconds between retries.

-comment - An optional comment about the snapshot.

-presnapshot - If false, this indicates that the snapshot that runs before the script should be skipped.

-postsnapshot - If false, this indicates that the snapshot that runs after the script should be skipped. If "task", this indicates that snapshot after the script should run as a separate task.

-priority - Task priority value (1, 2, 3, 4 or 5). Invalid priority will be changed to an appropriate value automatically.

Return Type

String

Examples

add vlan trunk -deviceid 2801 -portname xxx -nativevlanid 1 -addvlanids 12,13,14

annotate access

Modify the comments on or the display name of a device access record.

Synopsis

annotate access -id <Device access record ID> [-comment <Comment>] [-name <Name>] [-customname <Custom name>] [-customvalue <Custom value>]

Description

-id - Specifies a device access record. Think of this as a "device access record ID".
-comment - Additional information regarding the access record.
-name - An optional name for the access record.
-customname - The custom field name
-customvalue - The custom field value

Return Type

Status

Examples

annotate access -id 2 -comment "Device tainted at this point." -name "Intrusion detected"
annotate access -id 2 -customname TicketID - customvalue 5

annotate config

Add a comment to the specified config.

Synopsis

annotate config -id <Config ID> -comment <comment>

Description

Note that comments added by means of this command are not added to the config itself. They are stored seperately along with the config.

-id - The ID of the config on which you are commenting.
-comment - Additional information regarding the config.

Return Type

Status

Examples

annotate config -id 1754 -comment "north campus group template."

assign driver

Manually assign a driver to a device.

Synopsis

assign driver [-ip <IP address>] [-id <Device ID>] -name <Driver Name>

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-id - A valid device id
-name - A valid internal driver name, supported by system

Return Type

Status

Examples

assign driver -ip 192.0.2.10 -name CiscoIOSGenericNoLog
assign driver -id 70 -name CiscoIOSGenericNoLog

configure syslog

Configure a device to send syslog messages to the system's change detection facilites.

Synopsis

configure syslog [-ip <IP address>] [-group <Groupname>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-rep <Task repeat period>] [-sync] [-start <Task start date>] [-comment <Snapshop comment>] [-usesyslogrelay <IP address>]

Description

Have the system configure the specified device to send all syslog messages for the system's change detection facilites to function optimally to the system's syslog server. The configuration operation is atually a scheduled task.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-group - A valid group name. Do not use this option with -ip (exactly one of -ip or -group must be specified).
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-rep - (#min | #:# | #days | #weeks | #months) where # is a positive integer. #:# is hours:minutes--the two integers do not have to be the same. Do not use this option with -sync.
-sync - Indicates the command should return only after the Configure Syslog task is complete. Do not use this option with -rep or -start.
-start - YYYY:MM:DD:HH:mm. The first date on which the task will run.
-comment - An optional comment about the Configure Syslog task.
-usesyslogrelay - Indicates to the syslog configuration task that the device currently logs to syslog relay host. Supply this option if you wish to set up forwarding on that relay host rather than have the device log directly to the system's syslog server. The specified IP address is taken to be the IP address of the relay host.

Return Type

String

Examples

configure syslog -ip 192.0.2.10
configure syslog -ip 192.0.2.10 -usesyslogrelay blanka
configure syslog -host Zangief -start 2004:02:29:23:59 -rep 1weeks
configure syslog -ip 192.0.2.10 -sync
configure syslog -group mygroup

deactivate device

Mark a device as deactivated.

Synopsis

deactivate device [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID

Return Type

Status

Examples

deactivate device -host rtr5.vfm.lab

del access

Delete access records.

Synopsis

del access [-id <Device Access Record ID.>] [-cutoff <Date>]

Description

This command can delete a single access record when provided that record's id (via the option "-id"), or all access records prior to a given date (via the option "-cutoff"). Provide exactly one of "-id", "-cutoff". Note that deleting access records will cause all configs associated with the deleted access record to also be deleted.

-id - A device access record ID.
-cutoff - YYYY:MM:DD:HH:mm. All access records prior to this date will be deleted.

Return Type

Status

Examples

del access -id 6288
del access -cutoff 2004:02:29:23:59

del authentication

Delete all password information associated with the specified device, or delete an authentication rule from the database.

Synopsis

del authentication [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-loc <Location>] [-site <Site Partition>] [-rulename <Rule Name>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.: The device for which password information should be deleted.
-host - A valid hostname: The device for which password information should be deleted.
-fqdn - A valid Fully Qualified Domain Name: The device for which password information should be deleted.
-deviceid - A device ID
-loc - Location
-site - site of rules to be deleted
-rulename - anme fo rules to be deleted

Return Type

Status

Examples

del authentication -ip 192.0.2.10
del authentication -loc db -site SiteA -rulename arule

del cache

Delete cached information from a region.

Synopsis

del cache -region <REGION_NAME>

Description

Delete information from a region in the cache.

-region - Specify the name of the region in use

Return Type

None

Examples

del cache -region REGION_NAME

del device

Delete the specified device.

Synopsis

del device [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID

Return Type

Status

Examples

del device -ip 192.0.2.10
del device -ip "East Site:192.0.2.10"

del device context

Delete A Device Context

Synopsis

del device context -deviceid <Device ID> -contextvariables <Context Variables>

Description

For drivers that support it, remove a virtual context from a device

-deviceid - The device ID to remove a context from

-contextvariables - A comma separated list of driver specific variables and values that are required to delete a context on the given device. These variables can be found using the 'list device context variables' command.

Return Type

String

Examples

del device context -deviceid 749 -contextvariables "context_name=foo"

del device data

Delete device configuration and diagnostic data.

Synopsis

del device data [-id <Config ID>] [-cutoff <Date>]

Description

This command can delete a single device data block when provided that device data id (via. the option "-id"), or all device data prior to a given date (via the option "-cutoff"). Provide exactly one of "-id", "-cutoff".

-id - A config ID
-cutoff - YYYY:MM:DD:HH:mm. All configs prior to this date will be deleted.

Return Type

Status

Examples

del device data -id 866227436
del device data -cutoff 2004:02:29:23:59

del device from group

Delete a device from a device group.

Synopsis

del device from group [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] -group <Device group>

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-group - The name of the device group from which the device should be deleted.

Return Type

Status

Examples

del device from group -ip 192.0.2.10 -group tech-dev

del device relationship

Delete the relationship identified by the relationship id

Synopsis

del device relationship -relationshipid <Relationship ID>

Description

Delete the relationship identified by the relationship id

-relationshipid - Specify the Relationship ID for the relationship which you want to delete.

Return Type

String

Examples

del device relationship -relationshipid 101

del device template

Delete the specified device template.

Synopsis

del device template -templateid <Device ID>

Description

-templateid - A device template ID

Return Type

STATUS

Examples

del device template -templateid 3301

del drivers

Delete all drivers associated with a device.

Synopsis

del drivers [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID

Return Type

Status

Examples

del drivers -ip 192.0.2.10

del event

Delete the specified event.

Synopsis

del event -id <event ID>

Description

-id - A valid event id

Return Type

Status

Examples

del event -id 799

del group

Delete a group from the system.

Synopsis

del group -name <Name> -type <Type>

Description

Specify the group by both its name and type.

-name - The name of the group to be removed.
-type - The type of the group to be removed.

Return Type

Status

Examples

del group -name "border routers" -type "device"
del group -name "test user group" -type "user"

del group from parent group

Remove a device group from a parent device group.

Synopsis

del group from parent group -parent <Parent group name> -child <Child group name>

Description

-parent - Name of the parent group
-child - Name of the child group

Return Type

Status

Examples

del group from parent group -parent "North America" -child "Costa Rica NOC"

del ip

Delete the specified ip address.

Synopsis

del ip -ipvalue <Value> [-deviceip <Device IP address>] [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-ipvalue - The ip value a.b.c.d where 0 <= a,b,c,d <= 255

-deviceip - The device's ip address a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID

Return Type

Status

Examples

del ip -deviceip 192.0.2.10 -ipvalue 192.0.2.10
del ip -deviceid 1401 -ipvalue 192.0.2.10

del metadata

Delete a specific piece of custom data.

Synopsis

del metadata -metadataid <Metadata ID>

Description

-metadataid - ID of the custom data to delete

Return Type

Status

Examples

del metadata -metadataid 54535

del metadata field

Delete a custom data field and all data currently associated with that field.

Synopsis

del metadata field -fieldid <Field ID>

Description

-fieldid - ID of the custom data field to delete

Return Type

Status

Examples

del metadata field -fieldid 8394

del partition

Delete a partition from view.

Synopsis

del partition -name <Name>

Description

-name - The name of the partition to be removed.

Return Type

Status

Examples

del partition -name "Redmond Site"

del script

Delete an existing command script, advanced script, or diagnostic.

Synopsis

del script [-id <Script / Diagnostic ID>] [-name <Script / Diagnostic Name>] [-type <Script / Diagnostic Type>]

Description

Delete the indicated command script, advanced script, or diagnostic. Specify the script or diagnostic by ID or by a combination of name and type. If more than one name match occurs, then an error will be reported and you must specify the unique script by ID.

-id - ID of the desired script or diagnostic
-name - Name of the desired script or diagnostic
-type - Type of the desired script or diagnostic - may be command, advanced or diagnostic

Return Type

Status

Examples

del script -id 5
del script -name "Edit Port Duplex" -type command

del service type

Remove all service types from a device.

Synopsis

del service type [-ip <IP Address>] [-deviceid <Device ID>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-deviceid - device id

Return Type

Status

Examples

del service type -deviceid 3508

del session

Delete an interceptor log record.

Synopsis

del session -id <Interceptor log id>

Description

-id - Interceptor log ID

Return Type

Status

Examples

del session -id 5

del system message

Delete the specified system message.

Synopsis

del system message -id <System message ID>

Description

-id - A valid system message id

Return Type

Status

Examples

del system message -id 799

del task

Delete a task whether it has run or not.

Synopsis

del task -id <Task ID>

Description

-id - A task ID

Return Type

Status

Examples

del task -id 4321

del user

Delete a user from the system.

Synopsis

del user -u <User name>

Description

-u - The user name to be deleted

Return Type

Status

Examples

del user -u johnd

del user from group

Delete a user from a user group.

Synopsis

del user from group -u <Username> -g <User group name>

Description

-u - Username
-g - User group name

Return Type

Status

Examples

del user from group -u johnd -g "User Group 1"

del vlan

Delete a VLAN given a VlanID

Synopsis

del vlan [-deviceip <Device IP address>] [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] -vlanid <Vlan ID> [-start <Task start date>] [-rep <Task repeat period>] [-sync] [-sessionlog <true or false>] [-retrycount <Retry count>] [-retryinterval <Retry interval>] [-comment <Snapshop comment>] [-presnapshot <true or false>] [-postsnapshot <true, false or task>] [-priority <Task priority>]

Description

-deviceip - The device's ip address a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-host - A valid hostname

-fqdn - A valid Fully Qualified Domain Name

-deviceid - A device ID

-vlanid - Vlan ID to delete

-start - YYYY:MM:DD:HH:mm. The first date on which the task will run.

-rep - (#min | #:# | #days | #weeks | #months) where # is a positive integer. #:# is hours:minutes--the two integers do not have to be the same. Do not use this option with -sync.

-sync - Indicates the command should return only after the snapshot retrieval task is complete. Do not use this option with -rep or -start.

-sessionlog - If true a complete session log will be saved with this task.

-retrycount - The number of times to retry the task if it fails.

-retryinterval - The number of seconds between retries.

-comment - An optional comment about the snapshot.

-presnapshot - If false, this indicates that the snapshot that runs before the script should be skipped.

-postsnapshot - If false, this indicates that the snapshot that runs after the script should be skipped. If "task", this indicates that snapshot after the script should run as a separate task.

-priority - Task priority value (1, 2, 3, 4 or 5). Invalid priority will be changed to an appropriate value automatically.

Return Type

String

Examples

del vlan -deviceid 2801 -vlanid 123

del vlan trunk

Deletes a VLAN Trunk

Synopsis

del vlan trunk [-deviceip <Device IP address>] [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] -portname <Port Name> -nativevlanid <Native vlan ID> [-start <Task start date>] [-rep <Task repeat period>] [-sync] [-sessionlog <true or false>] [-retrycount <Retry count>] [-retryinterval <Retry interval>] [-comment <Snapshop comment>] [-presnapshot <true or false>] [-postsnapshot <true, false or task>] [-priority <Task priority>]-deviceip <Device IP address>

Description

The device's ip address a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-ip <IP address> a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-host <Hostname> A valid hostname

-fqdn <Fully Qualified Domain Name> A valid Fully Qualified Domain Name

-deviceid <Device ID> A device ID

-portname <Port Name>trunk port name to delete

-nativevlanid <Native vlan ID> specify a native or default vlan id

-start <Task start date> YYYY:MM:DD:HH:mm. The first date on which the task will run.

-rep <Task repeat period> (#min | #:# | #days | #weeks | #months) where # is a positive integer. #:# is hours:minutes--the two integers do not have to be the same. Do not use this option with -sync.

-sync Indicates the command should return only after the snapshot retrieval task is complete. Do not use this option with -rep or -start.

-sessionlog <true or false> If true a complete session log will be saved with this task.

-retrycount <Retry count> The number of times to retry the task if it fails.

-retryinterval <Retry interval> The number of seconds between retries.

-comment <Snapshop comment> An optional comment about the snapshot.

-presnapshot <true or false> If false, this indicates that the snapshot that runs before the script should be skipped.

-postsnapshot <true, false or task> If false, this indicates that the snapshot that runs after the script should be skipped. If "task", this indicates that snapshot after the script should run as a separate task.

-priority <Task priority> Task priority value (1, 2, 3, 4 or 5). Invalid priority will be changed to an appropriate value automatically.

Return Type

String

Examples

del vlan trunk -deviceid 2801 -portname xxx -nativevlanid 11

delete image

Delete software images from the database.

Synopsis

delete image [-site <partition site>] -imageset <imageset name> -images <images separated by ,>

Description

Delete software images from database

-site - the site which the image belongs to. Will assume to be global if not specified.
-imageset - imageset name the images will be deleted.
-images - images to be deleted.

Return Type

Status

Examples

delete image -site SiteA -imageset fooset -images bar.bin,baz.bin

deploy config

Deploy config to a device.

Synopsis

deploy config [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-id <Config ID>] [-configtext <Config Text>] [-start <Task start date>] [-sync] -option <Deployment option>

Description

Deploy the specified config to a specified device either right away or at some point in the future. The deploy operation is actually a scheduled task.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-id - The ID of the config to deploy to the specified device.
-configtext - The configuration text to deploy to the specified device.
-start - YYYY:MM:DD:HH:mm. The first date on which the task will run. Do not use this option with -sync.
-sync - Indicates that the command should return only after the deploy task is complete. Do not use this option with -start.
-option - current or startup_reload, as applicable to the device.

Return Type

String

Examples

deploy config -ip 192.0.2.10 -id 1962 -sync -option current
deploy config -ip "East Office:192.0.2.10" -id 1962 -sync -option current
deploy config -ip 192.0.2.10 -id 276 -start 2004:02:29:23:59 -option startup_reload
deploy config -ip 192.0.2.10 -configtext "logging 192.0.2.10\nlogging192.0.2.10" -option current

deploy image

Deploy software images to device.

Synopsis

deploy image -ip <device ip address> [-site <site of imageset>] -imageset <imageset name> -images <images separated by ,> [-reboot <reboot instruction>] [-rebootwait <reboot wait (in seconds)>] [-filesystem <file system of device>] [-pretask <task to run before deployment>] [-posttask <task to run after deployment>] [-verify <true|false>] [-start <Task start date>] [-comment <Snapshot comment>] [-duration <Estimated duration of snapshot task.>] [-sessionlog <true or false>] [-customname <Custom name>] [-customvalue <Custom value>] [-retryInterval <Retry count>] [-retryCount <Retry interval>]

Description

-ip - ip address of the device the images will deploy to.
-site - partition site which the imageset belongs to. Will look for global imagesets if not specified.
-imageset - imageset name the images from.
-images - images from the imageset to be deployed.
-reboot - wheather to reboot the device after deploy images.
-rebootwait - seconds to wait before reboot.

-filesystem - filesystem name of the device the images will deploy to.
-pretask - name of task before deployment.
-posttask - name of task after deployment.
-verify - verify the image after deployment.
-start - YYYY:MM:DD:HH:mm. The first date on which the task will run. The string "now" means the current time. The string "tomorrow" means 24 hours from the current time.
-comment - An optional comment about the snapshot.
-duration - A number concatenated with a units signifier. Valid signifiers are m (minutes), h (hours), d (days), w (weeks). If this option is not provided, the duration for the task is set to 60 minutes.
-sessionlog - If true a complete session log will be saved with this task.
-customname - The custom field name.
-customvalue - The custom field value.
-retryInterval - The number of seconds between retries.
-retryCount - The number of times to retry the task if it fails.

Return Type

String

Examples

deploy image -ip 192.0.2.10 -imageset fooset -images bar.bin,foo.bin -filesystem flash:
deploy image -ip 192.0.2.10 -imageset fooset -images bar.bin,foo.bin -filesystem flash: -reboot -rebootwait 60
deploy image -ip 192.0.2.10 -site SiteA -imageset fooset -images bar.bin,foo.bin -filesystem flash: -reboot -rebootwait 60 -posttask squeeze
deploy image -ip 192.0.2.10 -imageset fooset -images bar.bin,foo.bin -filesystem flash: -verify true

diff config

Show the differences between two configs.

Synopsis

diff config -id1 <Config ID> -id2 <Config ID>

Description

-id1 - The ID of a config
-id2 - The ID of a config

Return Type

String

Examples

diff config -id1 1961 -id2 1989

disable device

Mark a device as disabled.

Synopsis

disable device [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID

Return Type

Status

Examples

disable device -host rtr5.vfm.lab

discover driver

Discover a driver for a device.

Synopsis

discover driver [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

Attempts to match a driver to the specified device.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.: The device for which a driver should be discovered.
-host - A valid hostname: The device for which a driver should be discovered.
-fqdn - A valid Fully Qualified Domain Name: The device for which a driver should be discovered.
-deviceid - A device ID

Return Type

Status

Examples

discover driver -ip 192.0.2.10
discover driver -ip "East Site:192.0.2.10"

discover drivers

Discover drivers for all devices.

Synopsis

discover drivers [-noskip] [-group <Device group for drivers discovery>]

Description

Attempts to match a driver to each device that the system recognizes.

-noskip - do not skip devices with known drivers
-group - discover drivers for specified group

Return Type

String

Examples

discover drivers
discover drivers -noskip

enable device

Mark a device as enabled.

Synopsis

enable device [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID

Return Type

Status

Examples

enable device -ip 192.0.2.10
enable device -ip "East Site:192.0.2.10"

exit

Exit the system.

Synopsis

exit

Return Type

None

Examples

exit

export policy

Export configuration policies to a file.

Synopsis

export policy -filename <export file name> -policies <policies to be exported. Specify "all" to export all policies>

Description

The export filename must be the absolute path to the file and must be directly accessible by the system.

-filename - Specify the absolute path to the export file. If the path contains spaces, you must quote the argument.
-policies - Provide a list of comma separated policy IDs or names to be exported.

Return Type

Status

Examples

export policy -filename c:/policy_export.xml -policies all

export policy -filename c:/policy_export.xml -policies 2301

get external credential

Get user credentials for external services

Synopsis

get external credential -category <External redential category>

Description

Get the username and passwords used to log on toexternal applications/websites such as Cisco.com and its proxy server.If current user has not defined the external credentials, the return value is null.Otherwise, the return value is a result set that contains following clear text attributes: username password

-category - Specify the type of the credentail to get. Must be an integer value. Supported values: 0, cisco.com; 1, cisco.com proxy

Return Type

UserExternalCredentialVO with columns:

mCategory

mCreateDate

mLastModifiedDate

mPassword

mUserExternalCredentialID

mUserID

mUsername

Examples

get external credential -category 0

get snapshot

Get the config from a device.

Synopsis

get snapshot [-ip <IP address>] [-group <Groupname>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-rep <Task repeat period>] [-sync] [-start <Task start date>] [-comment <Snapshot comment>] [-duration <Estimated duration of snapshot task.>] [-sessionlog <true or false>] [-customname <Custom name>] [-customvalue <Custom value>] [-retryInterval <Retry count>] [-retryCount <Retry interval>]

Description

Get the config from a specified device either right away, or at some point in the future. The retrieval operation is actually a scheduled task. Using this command, you can set the task to repeat periodically.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-group - A valid group name. Do not use this option with -ip (exactly one of -ip or -group must be specified).
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-rep - (#min | #:# | #days | #weeks | #months) where # is a positive integer. #:# is hours:minutes--the two integers do not have to be the same. Do not use this option with -sync.
-sync - Indicates the command should return only after the snapshot retrieval task is complete. Do not use this option with -rep or -start.
-start - YYYY:MM:DD:HH:mm. The first date on which the task will run. The string "now" means the current time. The string "tomorrow" means 24 hours from the current time.
-comment - An optional comment about the snapshot.
-duration - A number concatenated with a units signifier. Valid signifiers are m (minutes), h (hours), d (days), w (weeks). If this option is not provided, the duration for the task is set to 60 minutes.
-sessionlog - If true a complete session log will be saved with this task.
-customname - The custom field name.
-customvalue - The custom field value.
-retryInterval - The number of seconds between retries.
-retryCount - The number of times to retry the task if it fails.

Return Type

String

Examples

```
get snapshot -ip 192.0.2.10
get snapshot -ip "East Office:192.0.2.10"
get snapshot -host Zangief -start 2004:02:29:23:59 -rep 2days
get snapshot -ip 192.0.2.10 -sync
get snapshot -group mygroup
```

import

Import device or device password information.

Synopsis

import -input <Filename or CSV data> -data <device or auth> [-log <Filename>] [-append <true or false>] [-discoverafter <true or false>] [-configuresyslog <true or false>] [-usesyslogrelay <Hostname>] [-filter <Filename>] [-cleanafter <true or false>] [-deviceorigin <Any String>] [-debug <true or false>]

Description

This command can import into the system device or device password information contained in appropriately formatted CSV files. Contact customer support for a CSV file format specification.

-input - Either the name of a file that contains CSV data or the CSV data itself.
-data - Whether the type of information imported is devices or device authentication.
-log - Command log file.
-append - If true, this command will append to the log file. If false, this command will overwrite the log file. This option is false by default.
-discoverafter - Discover drivers for imported device? This option is false by default.
-configuresyslog - Configure devices to send syslog messages to the system? Valid values are true | false
-usesyslogrelay - The name of a syslog relay host to use
-filter - An application that reads the input file from stdin, and writes a the system compatible CSV file to stdout.
-cleanafter - If true, then after importing data, a process will run on the server that will delete old devices. Devices are deleted according to the current configuration of the system's "deletion-on-import" rules, and the argument to the deviceorigin option. This option is false by default.
-deviceorigin - A description of the source of the data. This is recorded by the system, but is not visible via any UI.
-debug - Include debug information in log file.

Return Type

String

Examples

import -input devices.csv -data device -log import.log -append true -cleanafter false -deviceorigin "Border Routers" -filter prepro.exe
import -input auth.csv -data auth -log import.log
import -data device -input "primaryIPAddress\n1.2.3.4\n1.10.3.4"

import policy

Import configuration policies from a file.

Synopsis

import policy -filename <Import file name>

Description

The import filename must contain the absolute path to the file and must be directly accessible by the system.The import file is usually created by exporting policies.

-filename - Specify the absolute path to the import file. If the path contains spaces, you must quote the argument.

Return Type

Status

Examples

import policy -filename c:/policy_export.xml

list access

List all access records for a device.

Synopsis

list access [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-start <Date>] [-end <Date>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-start - Display only those access records created on or after the given date. Values for this option may be in one of the following formats:YYYY-MM-DD HH:MM:SS e.g. 2002-09-06 12:30:00YYYY-MM-DD HH:MM e.g. 2002-09-06 12:30YYYY-MM-DD e.g. 2002-09-06YYYY/MM/DD e.g. 2002/09/06YYYY:MM:DD:HH:MM e.g. 2002:09:06:12:30Or, one of: now, today, yesterday, tomorrowOr, in the format: " " e.g. "3 days ago" is a positive integer. is one of: seconds, minutes, hours, days, weeks, months, years;. is one of: ago, before, later, after.
-end - Display only those access records created on or before the given date. Values for this option have the same format as for the option -start.

Return Type

Collection:DeviceAccessLogVO with columns:

accessTrigger
actionTaken
changeEventData
comments
configComments
configID
createDate
createUserID
deviceAccessLogID
deviceDataCustom1
deviceDataCustom2
deviceDataCustom3
deviceDataCustom4
deviceDataCustom5
deviceDataCustom6
deviceID
displayName
externalChangeRequestID
interceptorLogID
isConfigChange
lastModifiedDate
noPrune

Examples

list access -ip 192.0.2.10
list access -ip "East Office:192.0.2.10"

list access all

List all access records for all devices.

Synopsis

list access all

Return Type

Collection:DeviceAccessLogVO with columns:

accessTrigger
actionTaken
changeEventData
comments
configComments
configID
createDate
createUserID
deviceAccessLogID
deviceDataCustom1

deviceDataCustom2
deviceDataCustom3
deviceDataCustom4
deviceDataCustom5
deviceDataCustom6
deviceID
displayName
externalChangeRequestID
interceptorLogID
isConfigChange
lastModifiedDate
noPrune

Examples

list access all

list acl

Lists all device ACLs in the system unless you include any of the options to limit the ACLs listed.

Synopsis

list acl [-host <Host Name>] [-ip <IP Address>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-aclid <ACL ID>] [-handle <Handle>] [-recent <Most Recent (true|false)>] [-includescript] [-includeapplication]

Description

-host - A valid host name.
-ip - List only ACLs with a valid IP Address (of format a.b.c.d where 0 <= a,b,c,d <= 255.)
-fqdn - List only ACLs with a valid Fully Qualified Domain Name
-deviceid - List only ACLs with this deviceid.
-aclid - List only ACLs with this aclid.
-handle - List only ACLs with this handle
-recent - Display only those acl's that are most recent.
-includescript - Include Script in the display.
-includeapplication - Include Application in the display.

Return Type

Collection:DeviceACLVO with columns:

aCLID
aCLType
application
applicationState
comments

deviceACLID
deviceID
handle
lastModifiedAccessLogID
lastModifiedDate
lastModifiedUserID
mostRecent
script

Examples

list acl
list acl -ip 192.0.2.10
list acl -ip 192.0.2.10 -aclid 139
list acl -ip 192.0.2.10 -deviceid 201
list acl -deviecid 501 -includescript
list acl -handle test34 -recent true -includeapplication

list all drivers

List all drivers installed in the system.

Synopsis

list all drivers

Description

Return Type

String

Examples

list all drivers

list authentication

Display authentication rules by rule name.

Synopsis

list authentication [-rulename <Rule Name>]

Description

-rulename - List Authentication by rule name

Return Type

Collection:AuthenticationRuleVO with columns:

authenticationRuleID
changeDate
comments
deviceAuthenticationID
deviceGroupID
endRange
lastModifiedUserID
ruleName
rulePattern
rulePriority
ruleType
siteID
siteName
startRange

Examples

list authentication -rulename 1

list basicip

List all configs for which the BasicIP model can be shown.

Synopsis

list basicip [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-start <Date>] [-end <Date>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-start - Display only those configs stored on or after the given date. Values for this option may be in one of the following formats:YYYY-MM-DD HH:MM:SS e.g. 2002-09-06 12:30:00YYYY-MM-DD HH:MM e.g. 2002-09-06 12:30YYYY-MM-DD e.g. 2002-09-06YYYY/MM/DD e.g. 2002/09/06YYYY:MM:DD:HH:MM e.g. 2002:09:06:12:30Or, one of: now, today, yesterday, tomorrowOr, in the format: e.g. 3 days ago is a positive integer. is one of: seconds, minutes, hours, days, weeks, months, years;. is one of: ago, before, later, after.
-end - Display only those configs stored on or before the given date. Values for this option have the same format as for the option -start.

Return Type

Collection:DeviceDataVO with columns:

blockFormat
blockSize
blockType
comments
createDate
customModel
dataBlock
deviceAccessLogID
deviceDataID
deviceID
lastModifiedDate
maskedSize
sourceDeviceDataID
variableData

Examples

list basicip -ip 192.0.2.10

list config

List all configs for the specified device.

Synopsis

list config [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-start <Date>] [-end <Date>] [-size] [-ids <Config ID List>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-start - Display only those configs stored on or after the given date. Values for this option may be in one of the following formats:YYYY-MM-DD HH:MM:SS e.g. 2002-09-06 12:30:00YYYY-MM-DD HH:MM e.g. 2002-09-06 12:30YYYY-MM-DD e.g. 2002-09-06YYYY/MM/DD e.g. 2002/09/06YYYY:MM:DD:HH:MM e.g. 2002:09:06:12:30Or, one of: now, today, yesterday, tomorrowOr, in the format: e.g. 3 days ago is a positive integer. is one of: seconds, minutes, hours, days, weeks, months, years;. is one of: ago, before, later, after.
-end - Display only those configs stored on or before the given date. Values for this option have the same format as for the option -start.
-size - Display the size (in bytes) of each config

-ids - List only configs in this comma-separated list of IDs.

Return Type

Collection:DeviceDataVO with columns:

blockFormat
blockSize
blockType
comments
createDate
customModel
dataBlock
deviceAccessLogID
deviceDataID
deviceID
lastModifiedDate
maskedSize
sourceDeviceDataID
variableData

Examples

list config -ip 192.0.2.10
list config -ip "East Office:192.0.2.10"
list config -ip 192.0.2.10 -size

list config all

List all configs for all devices.

Synopsis

list config all

Return Type

Collection:DeviceDataVO with columns:

blockFormat
blockSize
blockType
comments
createDate
customModel
dataBlock
deviceAccessLogID
deviceDataID
deviceID
lastModifiedDate
maskedSize

sourceDeviceDataID
variableData

Examples

list config all

list config id

List config IDs for the specified configs.

Synopsis

list config id [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-start <Date>] [-end <Date>] [-id <Config ID>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-start - Display only those configs stored on or after the given date. Values for this option may be in one of the following formats:YYYY-MM-DD HH:MM:SS e.g. 2002-09-06 12:30:00YYYY-MM-DD HH:MM e.g. 2002-09-06 12:30YYYY-MM-DD e.g. 2002-09-06YYYY/MM/DD e.g. 2002/09/06YYYY:MM:DD:HH:MM e.g. 2002:09:06:12:30Or, one of: now, today, yesterday, tomorrowOr, in the format: e.g. 3 days ago is a positive integer. is one of: seconds, minutes, hours, days, weeks, months, years;. is one of: ago, before, later, after.
-end - Display only those configs stored on or before the given date. Values for this option have the same format as for the option -start.
-id - Display only the specified config id.

Return Type

Collection:IntegerVO

Examples

list config id -ip 192.0.2.10
list config id -ip "East Site:192.0.2.10"

list custom data definition

List custom data definition by table name.

Synopsis

list custom data definition -tablename <Table Name>

Description

-tablename - List Custom Data for specific table

Return Type

Collection:CustomDataVO with columns:

columnName
customDataID
fieldLabel
fieldName
fieldValues
flags
inUse
lastModifiedDate
tableName

Examples

list custom data definition -tablename "Device Configuration & Diagnostics"
list custom data definition -tablename Devices
list custom data definition -tablename "Device Blades/Modules"
list custom data definition -tablename "Device Interfaces"
list custom data definition -tablename "Device Groups"
list custom data definition -tablename Users
list custom data definition -tablename "User Groups"
list custom data definition -tablename Tasks
list custom data definition -tablename "Telnet/SSH Sessions"

list device

Lists all devices in the system unless you include any of the options to limit the devices listed.

Synopsis

list device [-software <Software Version>] [-vendor <Device Vendor>] [-type <Device Type>] [-model <Device Model>] [-family <Device Family>] [-group <Device Group>] [-disabled] [-pollexcluded] [-ids <Device ID List>] [-hierarchy <Hierarchy Layer>] [-host <Device Host Name>] [-ip <Device IP Address>] [-realm <Realm Name>] [-startid <ID>] [-limitcount <Count>]

Description

-software - List only devices running this software
-vendor - List only devices with this vendor name
-type - List only devices of this type (Router, Switch, etc.)
-model - List only devices of this model ("2500 (3000 series)", BIG-IP, etc.)
-family - List only devices in this device family ("Cisco IOS", F5, etc.)

-group - List only devices in this device group
-disabled - List only devices that are unmanaged.
-pollexcluded - List only devices excluded from polling.
-ids - List only devices in this comma-separated list of IDs.
-hierarchy - List only devices in this hierarchy layer.
-host - List only devices with this host name
-ip - List only devices with this IP Address
-realm - List only devices in this realm
-startid - List devices starting with DeviceIDs greater than or equal to this one.
-limitcount - Return this many rows (maximum defaults to 10000).

Return Type

Collection:DeviceVO with columns:

accessMethods
assetTag
changeEventData
comments
consoleIPAddress
consolePort
consoleRealmName
contact
createDate
deviceCustom1
deviceCustom2
deviceCustom3
deviceCustom4
deviceCustom5
deviceCustom6
deviceGroup1ID
deviceGroup2ID
deviceGroup3ID
deviceID
deviceName
deviceType
driverName
duplexMismatchDetected
excludeFromPoll
feedSource
firmwareVersion
flashMemory
freePorts
geographicalLocation
hierarchyLayer
hostName
lastAccessAttemptDate
lastAccessAttemptStatus
lastAccessSuccessDate
lastConfigChangeUserID
lastConfigPolicyCheckedDate

lastDuplexDataUpdate
lastImportDate
lastModifiedUserID
lastRecordModifiedDate
lastSnapshotAttemptDate
lastSnapshotAttemptStatus
lastSnapshotDate
lastSnapshotSuccessDate
lastTopologyDataUpdate
lastUsedAuthenticationID
latestSoftwareHistoryID
latestStartupRunningDiffer
managementStatus
memory
model
modemNumber
mostRecentConfigID
nATIPAddress
nATRealmName
performACLParsing
policyImportance
policyInCompliance
primaryFQDN
primaryIPAddress
processor
putInServiceDate
rOMVersion
serialNumber
siteID
softwareVersion
tFTPServerIPAddress
ticketNumber
totalPorts
unmanagedDate
uptime
uptimeStoredDate
vendor

Examples

list device
list device -group "border routers"
list device -group "border routers" -disabled
list device -family "Cisco IOS"
list device -vendor Nortel
list device -ids 1023,763,8723

list device context variables

List device context variables.

Synopsis

list device context variables -deviceid <Device ID> -action <Action>

Description

Adding device contexts requires device specific parameters. This command lists the device context variables needed to perform a context addition or removal. For example, one device might require a context name and a config location, while a different device might require a context name and a slot number. This command will list what is needed for the device you are working on. This information is then used as input into the add device context command.

-deviceid - The deviceID to get context variable names for

-action - The action to get context variable names for (add or remove)

Return Type

String

Examples

list device context variables -deviceid 749 -action add

list device data

List configuration and diagnostic data records for the specified device.

Synopsis

list device data -ip <IP address> [-dataType <Data type>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-dataType - A string describing the type of device data record to list

Return Type

Collection:DeviceDataVO with columns:

blockFormat
blockSize
blockType
comments
createDate
customModel
dataBlock
deviceAccessLogID
deviceDataID
deviceID

lastModifiedDate
maskedSize
sourceDeviceDataID
variableData

Examples

list device data -ip 192.0.2.10
list device data -ip 192.0.2.10 -dataType "configuration"

list device family

List device families in the system.

Synopsis

list device family [-software <Software Version>] [-vendor <Device Vendor>] [-type <Device Type>] [-model <Device Model>] [-group <Device Group>]

Description

-software - List only device families for devices running this software
-vendor - List only device families for devices with this vendor name
-type - List only device families for devices of this type (Router, Switch, etc.)
-model - List only device families for devices of this model ("2500 (3000 series)", BIG-IP, etc.)
-group - List only device families for devices in this device group

Return Type

Collection:StringVO

Examples

list device family
list device family -group "border routers"
list device family -vendor Nortel

list device group

List device groups in the system.

Synopsis

list device group [-software <Software Version>] [-vendor <Device Vendor>] [-type <Device Type>] [-model <Device Model>] [-family <Device Family>] [-parent <Parent Device Group Name>]

Description

-software - List only device groups for devices running this software
-vendor - List only device groups for devices with this vendor name
-type - List only device groups for devices of this type (Router, Switch, etc.)
-model - List only device groups for devices of this model ("2500 (3000 series)", BIG-IP, etc.)
-family - List only device groups for devices in this device family ("Cisco IOS", F5, etc.)
-parent - List only device groups that are direct descendants of this parent device group name

Return Type

Collection:StringVO

Examples

list device group
list device group -family "Cisco IOS"
list device group -vendor Nortel

list device id

Lists all device IDs in the system unless you include any of the options to limit the device IDs listed.

Synopsis

list device id [-software <Software Version>] [-vendor <Device Vendor>] [-type <Device Type>] [-model <Device Model>] [-family <Device Family>] [-group <Device Group>] [-disabled] [-pollexcluded] [-id <Device ID>] [-host <Device Host Name>] [-ip <Device IP Address>] [-realm <Realm Name>] [-hierarchy <Hierarchy Layer>] [-viewable_by <Viewable By>]

Description

-software - List only devices running this software
-vendor - List only devices with this vendor name
-type - List only devices of this type (Router, Switch, etc.)
-model - List only devices of this model ("2500 (3000 series)", BIG-IP, etc.)
-family - List only devices in this device family ("Cisco IOS", F5, etc.)
-group - List only devices in this device group
-disabled - List only devices that are unmanaged.
-pollexcluded - List only devices excluded from polling.
-id - List only this device.
-host - List only devices with this host name
-ip - List only devices with this IP Address
-realm - List only devices in this realm
-hierarchy - List only devices with this hierarchy layer

-viewable_by - List only devices that are viewable by this user_ID

Return Type

Collection:IntegerVO

Examples

list device id
list device id -group "border routers"
list device id -group "border routers" -disabled
list device id -family "Cisco IOS"
list device id -vendor Nortel
list device id -viewable_by 201

list device model

List device model names in the system.

Synopsis

list device model [-software <Software Version>] [-vendor <Device Vendor>] [-type <Device Type>] [-family <Device Family>] [-group <Device Group>]

Description

-software - List only device model names for devices running this software
-vendor - List only device model names for devices with this vendor name
-type - List only device model names for devices of this type (Router, Switch, etc.)
-family - List only device model names for devices in this device family ("Cisco IOS", F5, etc.)
-group - List only device model names for devices in this device group

Return Type

Collection:StringVO

Examples

list device model
list device model -group "border routers"
list device model -family "Cisco IOS"
list device model -vendor Nortel

list device relationships

List device relationships.

Synopsis

list device relationships [-relationshipid <Relationship ID>] [-origindeviceid <Originating Device ID>] [-destdeviceid <Destination Device ID>] [-relationshiptypeid <Relationship Type ID>] [-createdby <Created By>]

Description

List device relationships based on the search criteria.

-relationshipid - Specify the relationship ID

-origindeviceid - Specify the Originating Device ID to list all the relationships where the given Device ID is originating device

-destdeviceid - Specify the Destination Device ID for the relationship where the given Device ID is destination device

-relationshiptypeid - Specifiy Realtionship type ID, 1: PC_USER , 2:PC_GENERIC , 3:PC_ADMIN_CONTEXT, 4:PEER_USER, 5:PEER_GENERIC

-createdby - Specify the type of creation for the relationship,i.e 1-Mnaual 2-System Inventory to filter the result

Return Type

Collection:DeviceRelationshipVO with columns:

createdBy
destinationDeviceID
deviceRelationshipID
lastModifiedDate
originatingDeviceID
relationshipTypeID

Examples

list device relationships -origindeviceid 281

list device relationships -destdeviceid 401

list device relationships -relationshiptypeid 2 -destdeviceid 391

list device software

List device software versions in the system.

Synopsis

list device software [-vendor <Device Vendor>] [-type <Device Type>] [-model <Device Model>] [-family <Device Family>] [-group <Device Group>]

Description

-vendor - List only device software versions for devices with this vendor name
-type - List only device software versions for devices of this type (Router, Switch, etc.)
-model - List only device software versions for devices of this model ("2500 (3000 series)", BIG-IP, etc.)
-family - List only device software versions for devices in this device family ("Cisco IOS", F5, etc.)
-group - List only device software versions for devices in this device group

Return Type

Collection:StringVO

Examples

list device software
list device software -group "border routers"
list device software -family "Cisco IOS"
list device software -vendor Nortel

list device template

List device templates.

Synopsis

list device template [-deviceip <Device IP>] [-deviceid <Device ID>] [-vendor <Device Vendor>] [-type <Device Type>] [-model <Device Model>] [-host <Device Host Name>]

Description

Lists all device templates in the system unless you include any of the options to limit the templates listed.

-deviceip - List all those templates applicable to that device for the given IP

-deviceid - List all those templates applicable to that device for the given device ID

-vendor - List only templates with this vendor name

-type - List only templates of this type (Router, Switch, etc.)

-model - List only templates of this model ("2500 (3000 series)", BIG-IP, etc.)

-host - List only templates with this host name

Return Type

Collection:DeviceVO with columns:

accessMethods
assetTag

changeEventData
comments
consoleIPAddress
consolePort
consoleRealmName
contact
createDate
deviceCustom1
deviceCustom2
deviceCustom3
deviceCustom4
deviceCustom5
deviceCustom6
deviceGroup1ID
deviceGroup2ID
deviceGroup3ID
deviceID
deviceName
deviceType
driverName
duplexMismatchDetected
excludeFromPoll
feedSource
firmwareVersion
flashMemory
freePorts
geographicalLocation
hierarchyLayer
hostName
lastAccessAttemptDate
lastAccessAttemptStatus
lastAccessSuccessDate
lastConfigChangeUserID
lastConfigPolicyCheckedDate
lastDuplexDataUpdate
lastImportDate
lastModifiedUserID
lastRecordModifiedDate
lastSnapshotAttemptDate
lastSnapshotAttemptStatus
lastSnapshotDate
lastSnapshotSuccessDate
lastTopologyDataUpdate
lastUsedAuthenticationID
latestSoftwareHistoryID
latestStartupRunningDiffer
managementStatus
memory

model
modemNumber
mostRecentConfigID
nATIPAddress
nATRealmName
performACLParsing
policyImportance
policyInCompliance
primaryFQDN
primaryIPAddress
processor
putInServiceDate
rOMVersion
realmName
serialNumber
siteID
siteName
softwareVersion
tFTPServerIPAddress
ticketNumber
totalPorts
unmanagedDate
uptime
uptimeStoredDate
vendor

Examples

list device template

list device template -deviceip 127.0.0.1

list device template -deviceid 601

list device template -vendor Nortel

list device template -model Nortel

list device template -host SomeTemplate

list device type

List device types in the system.

Synopsis

list device type [-software <Software Version>] [-vendor <Device Vendor>] [-model <Device Model>] [-family <Device Family>] [-group <Device Group>]

Description

-software - List only device types for devices running this software
-vendor - List only device types for devices with this vendor name
-model - List only device types for devices of this model ("2500 (3000 series)", BIG-IP, etc.)
-family - List only device types for devices in this device family ("Cisco IOS", F5, etc.)
-group - List only device types for devices in this device group

Return Type

Collection:StringVO

Examples

list device type
list device type -group "border routers"
list device type -family "Cisco IOS"
list device type -vendor Nortel

list device vendor

List device manufacturers in the system.

Synopsis

list device vendor [-software <Software Version>] [-type <Device Type>] [-model <Device Model>] [-family <Device Family>] [-group <Device Group>]

Description

-software - List only device manufacturers for devices running this software
-type - List only device manufacturers for devices of this type (Router, Switch, etc.)
-model - List only device manufacturers for devices of this model ("2500 (3000 series)", BIG-IP, etc.)
-family - List only device manufacturers for devices in this device family ("Cisco IOS", F5, etc.)
-group - List only device manufacturers for devices in this device group

Return Type

Collection:StringVO

Examples

list device vendor
list device vendor -group "border routers"
list device vendor -family "Cisco IOS"

list device vtp

Lists all Devices in a particular VTP domain

Synopsis

list device vtp -domain <Domain name>

Description

-domain - Domain name

Return Type

Collection:DeviceVTPVO with columns:

configurationVersion
currentVLANNumber
description
deviceID
deviceVTPID
domainName
lastModifiedDate
lastModifiedUserID
maxVLANNumber
md5Digest
operatingMode
pruningMode
vtpTrapsGeneration
vtpV2Mode
vtpVersion

Examples

list device vtp -domain RnDlab

list deviceinfo

List all configs for which the DeviceInformation model can be shown.

Synopsis

list deviceinfo [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name

-deviceid - A device ID

Return Type

Collection:DeviceDataVO with columns:

blockFormat
blockSize
blockType
comments
createDate
customModel
dataBlock
deviceAccessLogID
deviceDataID
deviceID
lastModifiedDate
maskedSize
sourceDeviceDataID
variableData

Examples

list deviceinfo -ip 192.0.2.10

list diagnostic

List all configs for which the given diagnostic may be shown.

Synopsis

list diagnostic -diagnostic <Diagnostic Name> [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-start <Date>] [-end <Date>]

Description

-diagnostic - A diagnostic name
-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-start - Display only those diagnostics stored on or after the given date. Values for this option may be in one of the following formats:YYYY-MM-DD HH:MM:SS e.g. 2002-09-06 12:30:00YYYY-MM-DD HH:MM e.g. 2002-09-06 12:30YYYY-MM-DD e.g. 2002-09-06YYYY/MM/DD e.g. 2002/09/06YYYY:MM:DD:HH:MM e.g. 2002:09:06:12:30Or, one of: now, today, yesterday, tomorrowOr, in the format: e.g. 3 days ago is a positive integer. is one of: seconds, minutes, hours, days, weeks, months, years;. is one of: ago, before, later, after.

-end - Display only those diagnostics created on or before the given date. Values for this option have the same format as for the option -start.

Return Type

Collection:DeviceDataVO with columns:

blockFormat
blockSize
blockType
comments
createDate
customModel
dataBlock
deviceAccessLogID
deviceDataID
deviceID
lastModifiedDate
maskedSize
sourceDeviceDataID
variableData

Examples

list diagnostic -ip 192.0.2.10 -diagnostic "vlan report"

list drivers

List all drivers associated with a device.

Synopsis

list drivers [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID

Return Type

Collection:DriverLookupVO with columns:

baseModelName
deviceFamilyName
deviceID

driverLookupID
driverName
isLookupSkipped
lastModifiedDate

Examples

list drivers -ip 192.0.2.10

list event

List all events and system messages.

Synopsis

list event [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-type <Event Type>] [-start <Date>] [-end <Date>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.: Display only those events associated with the specified device.
-host - A valid hostname: Display only those events associated with the specified device.
-fqdn - A valid Fully Qualified Domain Name: Display only those events associated with the specified device.
-deviceid - A device ID
-type - A valid event type: Display only events of this type.Values for this option may one of the following:Approval No Longer RequiredApproval RequestApproval GrantedApproval Task ChangedApproval Task DeletedApproval DeniedApproval Task TimeoutApproval OverrideCommand Authorization ErrorCommand Script ModifiedUser Authentication ErrorConfiguration Policy AddedConfiguration Policy Non-ComplianceConfiguration Policy ChangedConfiguration Policy Pattern TimeoutConfiguration Rule AddedConfiguration Rule ChangedDevice Access FailureDevice AddedDevice Password ChangeDevice BootedDevice Command Script FailedDevice Command Script Completed SuccessfullyDevice Configuration ChangeDevice Configuration Change - No UserDevice Configuration Deployment FailureDevice Configuration DeploymentDevice Data FailureDevice DeletedDevice Diagnostic ChangedDevice Diagnostic FailedDevice Diagnostic Completed SuccessfullyDevice Flash Storage Running LowGroup ModifiedGroup AddedGroup DeletedDevice InaccessibleDevice EditedLast Used Device Password ChangedDevice ManagedDevice Missing from ImportDevice Permissions - ModifiedDevice Reservation ConflictDevice SnapshotDevice Software ChangeDevice Startup/Running Config DifferenceDevice UnmanagedDiagnostic ModifiedSoftware Vulnerability DetectedEmail Report SavedExternal Directory Server Authentication ErrorLicense Almost ExceededLicense Almost ExpiredLicense ExceededLicense ExpiredModule AddedModule ChangedModule RemovedMonitor OkayMonitor

ErrorDevice Permissions - New DeviceDevice Password Change FailureConcurrent Telnet/SSH Session OverrideReserved Device Configuration ChangedScheduled for Deploy Configuration EditedScheduled for Deploy Password ModifiedServer StartupSession Data CapturedSoftware Update FailedSoftware Update SucceededSummary Reports GeneratedPending Task DeletedTask StartedTicket CreatedUser LoginUser LogoutUser AddedUser DeletedUser Permission ChangedUser Message
-start - Display only events after this date.Values for this option may be in one of the following formats:YYYY-MM-DD HH:MM:SS e.g. 2002-09-06 12:30:00YYYY-MM-DD HH:MM e.g. 2002-09-06 12:30YYYY-MM-DD e.g. 2002-09-06YYYY/MM/DD e.g. 2002/09/06YYYY:MM:DD:HH:MM e.g. 2002:09:06:12:30Or, one of: now, today, yesterday, tomorrowOr, in the format: e.g. 3 days ago is a positive integer. is one of: seconds, minutes, hours, days, weeks, months, years;. is one of: ago, before, later, after.
-end - Display only events before this date.

Return Type

Collection:EventVO with columns:

configPolicyID
eventClass
eventData
eventDate
eventDeviceID
eventID
eventTaskID
eventText
eventType
eventUserID
importance
lastModifiedDate
siteID
ticketNumber

Examples

list event -ip 192.0.2.10
list event -ip "East Site:192.0.2.10"
list event -start yesterday

list group id

List all device groups or user groups viewable by a particular user.

Synopsis

list group id -type <Type> [-viewable_by <Viewable By>]

Description

-type - Type
-viewable_by - List only groups that are viewable by this user_ID

Return Type

Collection:IntegerVO

Examples

list group id -type device -viewable_by 201
list group id -type user -viewable_by 201

list groups

List groups of the specified type for a specific device or all groups in the system.

Synopsis

list groups -type <Type> [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-parent <Parent Group Name>]

Description

-type - The type of the groups to be listed. "device" is currently the only valid argument to this option.
-ip - List all device groups containing the device with this IP address
-host - List all device groups containing the device with this hostname
-fqdn - List all device groups containing the device with this Fully Qualified Domain Name
-deviceid - List all device groups containing the device with this device ID
-parent - List all device groups that are children of the indicated parent group

Return Type

Collection:DeviceGroupVO with columns:

comments
core
createDate
deviceCount
deviceGroup1ID
deviceGroup2ID
deviceGroup3ID
deviceGroupCustom1
deviceGroupCustom2
deviceGroupCustom3
deviceGroupCustom4
deviceGroupCustom5

deviceGroupCustom6
deviceGroupID
deviceGroupName
deviceViewID
devices
dynamicFilter
isDynamic
isParent
isPolicyScope
lastModifiedDate
managerUserID
managingCoreID
parentDeviceGroupID
realmName
shared

Examples

list groups -type device
list groups -type device -ip 192.0.2.10
list groups -type device -parent "North America"

list icmp

List all configs for which the ICMPTest model may be shown.

Synopsis

list icmp [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-start <Date>] [-end <Date>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-start - Display only those ICMPTest models stored on or after the given date. Values for this option may be in one of the following formats:YYYY-MM-DD HH:MM:SS e.g. 2002-09-06 12:30:00YYYY-MM-DD HH:MM e.g. 2002-09-06 12:30YYYY-MM-DD e.g. 2002-09-06YYYY/MM/DD e.g. 2002/09/06YYYY:MM:DD:HH:MM e.g. 2002:09:06:12:30Or, one of: now, today, yesterday, tomorrowOr, in the format: e.g. 3 days ago is a positive integer. is one of: seconds, minutes, hours, days, weeks, months, years;. is one of: ago, before, later, after.
-end - Display only those ICMPTest models stored on or before the given date. Values for this option have the same format as for the option -start.

Return Type

Collection:DeviceDataVO with columns:

blockFormat
blockSize
blockType
comments
createDate
customModel
dataBlock
deviceAccessLogID
deviceDataID
deviceID
lastModifiedDate
maskedSize
sourceDeviceDataID
variableData

Examples

list icmp -ip 192.0.2.10

list image

List images in database or device.

Synopsis

list image [-ip <ip address>] [-site <partition site>] [-imageset <imageset name>]

Description

Use -imageset option to list images in database and -ip to list images in device.

-ip - The device ip which the images list from.
-site - The partition site which the image belongs to. Will assume global if not specified.
-imageset - The imageset which images list from.

Return Type

Collection:DeviceSoftwareImageVO with columns:

bootROMRequired
checksum
comments
createDate
createType
createUserID

deviceSoftwareImageID
driverRequired
flashRAMRequired
hardwareRequired
imageName
imagePath
imageSetName
imageSize
lastModifiedDate
modelRequired
siteID
volatileRAMRequired

Examples

list image -ip 10.1.1.1
list image -site SiteA -imageset fooset

list imageoption

List information for device which is not under management, but in configuration data for software image management purpose.

Synopsis

list imageoption -name <device property name (model|processor|bootrom)>

Description

-name - device property name to list.

Return Type

Collection:StringVO

Examples

list imageoption -name model

list imageset

List imageset in database.

Synopsis

list imageset [-site <partition site>]

Description

-site - The partition site which the imageset belongs to. Will assume global if not specified.

Return Type

Collection:SoftwareImageSetVO with columns:

imageSetName
siteName

Examples

list imageset
list imageset -site SiteA

list int

List all configs for which the ShowInterfaces model may be shown.

Synopsis

list int [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-start <Date>] [-end <Date>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-start - Display only those ShowInterfaces models stored on or after the given date. Values for this option may be in one of the following formats:YYYY-MM-DD HH:MM:SS e.g. 2002-09-06 12:30:00YYYY-MM-DD HH:MM e.g. 2002-09-06 12:30YYYY-MM-DD e.g. 2002-09-06YYYY/MM/DD e.g. 2002/09/06YYYY:MM:DD:HH:MM e.g. 2002:09:06:12:30Or, one of: now, today, yesterday, tomorrowOr, in the format: e.g. 3 days ago is a positive integer. is one of: seconds, minutes, hours, days, weeks, months, years;. is one of: ago, before, later, after.
-end - Display only those ShowInterfaces models stored on or before the given date. Values for this option have the same format as for the option -start.

Return Type

Collection:DeviceDataVO with columns:

blockFormat
blockSize

blockType
comments
createDate
customModel
dataBlock
deviceAccessLogID
deviceDataID
deviceID
lastModifiedDate
maskedSize
sourceDeviceDataID
variableData

Examples

list int -ip 192.0.2.10

list ip

Lists ip addresses for specific device.

Synopsis

list ip [-deviceip <Device IP address>] [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-deviceip - The device's ip address a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID

Return Type

Collection:IpVO with columns:

changeDate
comments
deviceID
devicePortID
ipCustom1
ipCustom2
ipCustom3
ipCustom4
ipCustom5

ipCustom6
ipID
ipMask
ipName
ipPriority
ipType
ipValue
lastModifiedUserID
networkAddressStart
paddedIp
password
realmName
redirectLocation
subnetBitCount
usedToAccess
userName
variables

Examples

list ip -deviceip 192.0.2.10
list ip -deviceid 1401

list ip all

List all secondary ip addresses in the system.

Synopsis

list ip all

Return Type

Collection:IpVO with columns:

changeDate
comments
deviceID
devicePortID
ipCustom1
ipCustom2
ipCustom3
ipCustom4
ipCustom5
ipCustom6
ipID
ipMask
ipName
ipPriority
ipType

ipValue
lastModifiedUserID
networkAddressStart
paddedIp
password
realmName
redirectLocation
subnetBitCount
usedToAccess
userName
variables

Examples

list ip all

list metadata

List all custom data associated with a specific entry in a specific table.

Synopsis

list metadata -table <Database Table> -associatedtableid <Matching Row ID>

Description

-table - Table the data is associated with
-associatedtableid - ID of the associated row from the table.

Return Type

Collection:MetadataVO with columns:

associatedTableID
data
fieldID
lastModifiedDate
metadataID

Examples

list metadata -table RN_DEVICE -associatedtableid 21031
list metadata -table RN_DEVICE_PORT -associatedtableid 221

list metadata field

List all custom data fields associated with a specific table.

Synopsis

list metadata field -table <Database Table>

Description

-table - Table the fields are associated with

Return Type

Collection:MetadataFieldVO with columns:

associatedTable
fieldDataType
fieldID
fieldName
fieldValues
flags
inUse
lastModifiedDate

Examples

list metadata field -table RN_DEVICE
list metadata field -table RN_DEVICE_PORT

list module

List modules (or blades) in the system.

Synopsis

list module [-model <Model Number>] [-type <Module Description>] [-firmware <Firmware Version>] [-hardware <Hardware Revision>] [-memory <Memory>] [-comment <Comment>] [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-group <Device Group Name>]

Description

-model - List only device modules matching this model number
-type - List only device modules matching this module description
-firmware - List only device modules matching this firmware version
-hardware - List only device modules matching this hardware revision
-memory - List only device modules with this amount of memory
-comment - List only device modules matching this comment
-ip - List only device modules on the device with this IP address
-host - List only device modules on the device with this hostname
-fqdn - List only device modules on the device with this Fully Qualified Domain Name

-deviceid - List only device modules on the device with this device ID
-group - List only device modules on all devices with this device group name

Return Type

Collection:DeviceModuleVO with columns:

comments
deviceID
deviceModuleID
firmwareVersion
hardwareRevision
lastModifiedDate
memory
moduleCustom1
moduleCustom2
moduleCustom3
moduleCustom4
moduleCustom5
moduleCustom6
moduleDescription
moduleModel
moduleOS
serialNumber
slot
slotNumber

Examples

list module -host border7.red
list module -type ethernet

list ospfneighbor

List all configs for which the ShowOSPFNeighbors model may be shown.

Synopsis

list ospfneighbor [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-start <Date>] [-end <Date>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-start - Display only those ShowOSPFNeighbors models stored on or after the given date. Values for this option may be in one of the following formats:YYYY-

MM-DD HH:MM:SS e.g. 2002-09-06 12:30:00YYYY-MM-DD HH:MM e.g. 2002-09-06 12:30YYYY-MM-DD e.g. 2002-09-06YYYY/MM/DD e.g. 2002/09/06YYYY:MM:DD:HH:MM e.g. 2002:09:06:12:30Or, one of: now, today, yesterday, tomorrowOr, in the format: e.g. 3 days ago is a positive integer. is one of: seconds, minutes, hours, days, weeks, months, years;. is one of: ago, before, later, after.
-end - Display only those ShowOSPFNeighbors models stored on or before the given date. Values for this option have the same format as for the option -start.

Return Type

Collection:DeviceDataVO with columns:

blockFormat
blockSize
blockType
comments
createDate
customModel
dataBlock
deviceAccessLogID
deviceDataID
deviceID
lastModifiedDate
maskedSize
sourceDeviceDataID
variableData

Examples

list ospfneighbor -ip 192.0.2.10

list partition

Show details for a single parition.

Synopsis

list partition -viewname <View Name>

Description

-viewname - The View Name to show.

Return Type

Collection:DeviceGroupVO with columns:

comments
core

createDate
deviceCount
deviceGroup1ID
deviceGroup2ID
deviceGroup3ID
deviceGroupCustom1
deviceGroupCustom2
deviceGroupCustom3
deviceGroupCustom4
deviceGroupCustom5
deviceGroupCustom6
deviceGroupID
deviceGroupName
deviceViewID
devices
dynamicFilter
isDynamic
isParent
isPolicyScope
lastModifiedDate
managerUserID
managingCoreID
parentDeviceGroupID
realmName
shared

Examples

list partition -viewname Site

list policy id

Lists IDs of all policies that apply to a given device.

Synopsis

list policy id -deviceid <Device ID>

Description

-deviceid - device id

Return Type

Collection:IntegerVO

Examples

list policy id -deviceid 312

list policy rule

List all rules of a policy.

Synopsis

list policy rule -policyid <Policy ID>

Description

-policyid - policy id

Return Type

Collection:ConfigRuleVO with columns:

appliesToEntireDeviceFamily
blockEndPattern
blockStartPattern
comments
conditions
configPolicyID
configRuleID
configRuleName
createDate
description
deviceFamily
evaluationLogic
importance
inUse
lastModifiedDate
lastModifiedUserID
ruleType
scope
ticketNumber

Examples

list policy rule -policyid 6120

list port

List ports (or interfaces) for a specific device in the system.

Synopsis

list port [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-ip - List all device ports on the device with this IP address
-host - List all device ports on the device with this hostname
-fqdn - List all device ports on the device with this Fully Qualified Domain Name
-deviceid - List all device ports on the device with this device ID

Return Type

Collection:DevicePortVO with columns:

associatedVlanID
comments
configuredDuplex
configuredSpeed
description
deviceID
devicePortID
ipAddresses
lastModifiedDate
macAddress
negotiatedDuplex
negotiatedSpeed
portAllows
portCustom1
portCustom2
portCustom3
portCustom4
portCustom5
portCustom6
portName
portState
portStatus
portType
slotNumber
temporaryVlanName

Examples

list port -host border7.red
list port -ip 192.0.2.10

list port channels

Lists all port channels on a specific device

Synopsis

list port channels [-deviceip <Device IP address>] [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-deviceip - The device's ip address a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-host - A valid hostname

-fqdn - A valid Fully Qualified Domain Name

-deviceid - A device ID

Return Type

Collection:PortChannelVO with columns:

aggregatedPorts
portChannelID
portChannelName

Examples

list port channels -deviceid 201

list relationship type

Lists the relationship type details.

Synopsis

list relationship type [-id <Relationship type ID>] [-modifiable <Modifiable>]

Description

Lists all the available relationship types in the system.

-id - Specify the relationship type ID , 1: PC_USER , 2:PC_GENERIC , 3:PC_ADMIN_CONTEXT, 4:PEER_USER, 5:PEER_GENERIC

-modifiable - Flag to list the relationship types which can be modifiable by the user i.e filters the system defined relationship types.

Return Type

Collection:RelationshipTypeVO with columns:

description
lastModifiedDate
modifiable
relationshipTypeID
relationshipTypeName

Examples

list relationship type

list relationship type -id 2

list relationship type -modifiable

list relationships for device

Lists all the relationships defined for a device in the system.

Synopsis

list relationships for device -ip <IP address> [-hostname <Hostname>] [-domain <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-relationshiptypeids <Relationship IDs>] [-createdby <Created By>] [-filteroriginating <Filter Orginating>] [-filterdestination <Filter Destination>] [-modifiable <Modifiable>]

Description

Lists the different relationships that a device is participating in. The arguments will allow to filter the results.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-hostname - A valid hostname

-domain - A valid Fully Qualified Domain Name

-deviceid - A valid device ID

-relationshiptypeids - A list of relationship type IDs separated by comma

-createdby - How this relation ship is created 1-System 2-Manual

-filteroriginating - List the relationships where the given device is the originating device

-filterdestination - List the relationships where the given device is the destination device

-modifiable - Flag to list the relationship types which can be modifiable by the user i.e filters the system defined relationship types.

Return Type

Collection:DeviceRelationshipVO with columns:

createdBy
destinationDeviceID
deviceRelationshipID
lastModifiedDate
originatingDeviceID
relationshipTypeID

Examples

list relationships for device -ip 10.255.12.36

list relationships for device -deviceid 271

list relationships for device -deviceid 271 -filteroriginating -modifiable

list relationships for device -deviceid 271 -filterdestination

list relationships for device -deviceid 271 -filteroriginating –modifiable

list routing

List all routing tables for a device.

Synopsis

list routing [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-start <Date>] [-end <Date>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-start - Display only those routing tables stored on or after the given date. Values for this option may be in one of the following formats:YYYY-MM-DD HH:MM:SS e.g. 2002-09-06 12:30:00YYYY-MM-DD HH:MM e.g. 2002-09-06 12:30YYYY-MM-DD e.g. 2002-09-06YYYY/MM/DD e.g. 2002/09/06YYYY:MM:DD:HH:MM e.g. 2002:09:06:12:30Or, one of: now, today, yesterday, tomorrowOr, in the format: e.g. 3 days ago is a positive integer. is one of: seconds, minutes, hours, days, weeks, months, years;. is one of: ago, before, later, after.
-end - Display only those routing tables stored on or before the given date. Values for this option have the same format as for the option -start.

Return Type

Collection:DeviceDataVO with columns:

blockFormat
blockSize
blockType
comments
createDate
customModel
dataBlock
deviceAccessLogID
deviceDataID
deviceID
lastModifiedDate
maskedSize
sourceDeviceDataID
variableData

Examples

list routing -ip 192.0.2.10
list routing -ip "East Site:192.0.2.10"

list rule condition

Lists all conditions of a policy rule.

Synopsis

list rule condition -ruleid <Rule ID>

Description

-ruleid - rule id

Return Type

Collection:RuleConditionVO with columns:

comments
configRuleID
createDate
exactOrder
lastModifiedDate
modelDataName
operand
operatorName
ruleConditionID
ruleConditionOrder
useRegexp

Examples

list rule condition -ruleid 6120

list script

List command scripts, advanced scripts, and/or diagnostics.

Synopsis

list script [-type <Type>] [-scripttype <Script Type>] [-name <Name>] [-mode <Mode>] [-ids <Script ID List>] [-sitename <Site Name>]

Description

-type - Type of the desired script or diagnostic - may be command, advanced or diagnostic
-scripttype - User defined script type (i.e. subcategory) - applies only to command scripts and advanced scripts
-name - Script name
-mode - Script mode - for command scripts and diagnostics the script's level of device access (such as Cisco IOS enable); for advanced scripts the device family (such as Cisco IOS)
-ids - List only scripts in this comma-separated list of IDs.
-sitename - Site Name of the site the script belongs to.

Return Type

Collection:CustomScriptVO with columns:

createDate
createUserID
customScriptID
description
lastModifyDate
lastModifyUserID
name
parameters
script
scriptMode
scriptType
siteID
taskType
variableData

Examples

list script
list script -type diagnostic
list script -type advanced -scripttype "Core Provisioning Scripts"
list script -name "Set Banner"
list script -mode "Cisco IOS enable"
list script -sitename "Default Site"

list script id

List command script IDs, advanced scripts, and/or diagnostics.

Synopsis

list script id [-type <Type>] [-scripttype <Script Type>] [-name <Name>] [-mode <Mode>] [-id <ID>]

Description

-type - Type of the desired script or diagnostic - may be command, advanced or diagnostic
-scripttype - User defined script type (i.e. subcategory) - applies only to command scripts and advanced scripts
-name - Script name
-mode - Script mode - for command scripts and diagnostics the script's level of device access (such as Cisco IOS enable); for advanced scripts the device family (such as Cisco IOS)
-id - Script ID

Return Type

Collection:IntegerVO

Examples

list script id
list script id -type diagnostic
list script id -type advanced -scripttype "Core Provisioning Scripts"
list script id -name "Set Banner"
list script id -mode "Cisco IOS enable"

list script mode

List valid modes for commands scripts and diagnostics, and valid device families for advanced scripts, for all devices or a specified device.

Synopsis

list script mode [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-id <Device ID>] [-type <Type>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-id - A device ID
-type - Type of the desired script or diagnostic - may be command, advanced or diagnostic

Return Type

Collection:StringVO

Examples

list script mode
list script mode -type diagnostic
list script mode -ip 192.0.2.10
list script mode -id 1420 -type advanced

list session

List all interceptor log records for a device.

Synopsis

list session [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-start <Date>] [-end <Date>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-start - Display only those interceptor log records created on or after the given date. Values for this option may be in one of the following formats:YYYY-MM-DD HH:MM:SS e.g. 2002-09-06 12:30:00YYYY-MM-DD HH:MM e.g. 2002-09-06 12:30YYYY-MM-DD e.g. 2002-09-06YYYY/MM/DD e.g. 2002/09/06YYYY:MM:DD:HH:MM e.g. 2002:09:06:12:30Or, one of: now, today, yesterday, tomorrowOr, in the format: e.g. 3 days ago is a positive integer. is one of: seconds, minutes, hours, days, weeks, months, years;. is one of: ago, before, later, after.
-end - Display only those interceptor log records created on or before the given date. Values for this option have the same format as for the option -start.

Return Type

Collection:InterceptorLogVO with columns:

deviceID
deviceIP
endDate
interceptorLogCustom1
interceptorLogCustom2
interceptorLogCustom3
interceptorLogCustom4
interceptorLogCustom5
interceptorLogCustom6
interceptorLogID
lastModifiedDate
m_userName

sessionData
sessionType
startDate
status
userID

Examples

list session -ip 192.0.2.10

list site

List all sites in the system.

Synopsis

list site

Description

Result includes the name of each site in the system and the number of devices in each site.

Return Type

Collection:SiteVO with columns:

m_comments
m_deviceGroupID
m_lastModifiedDate
m_managingCoreID
m_name
m_owningCoreID
m_realmName
m_siteID

Examples

list site

list sys oids all

List all sys oids in the system.

Synopsis

list sys oids all

Return Type

Collection:StringVO

Examples

list sys oids all

list system message

List all system messages.

Synopsis

list system message [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-start <Date>] [-end <Date>]

Description

Lists all system messages unless you include one of the options. Including one of the device options displays all system messages associated with the specified device.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-start - Display only those system messages created on or after the given date. Values for this option may be in one of the following formats:YYYY-MM-DD HH:MM:SS e.g. 2002-09-06 12:30:00YYYY-MM-DD HH:MM e.g. 2002-09-06 12:30YYYY-MM-DD e.g. 2002-09-06YYYY/MM/DD e.g. 2002/09/06YYYY:MM:DD:HH:MM e.g. 2002:09:06:12:30Or, one of: now, today, yesterday, tomorrowOr, in the format: e.g. 3 days ago is a positive integer. is one of: seconds, minutes, hours, days, weeks, months, years;. is one of: ago, before, later, after.
-end - Display only those system messages created on or before the given date. Values for this option have the same format as for the option -start.

Return Type

Collection:EventVO with columns:

configPolicyID
eventClass
eventData
eventDate
eventDeviceID
eventID
eventTaskID
eventText
eventType

eventUserID
importance
lastModifiedDate
siteID
ticketNumber

Examples

list system message
list system message -host chi-border-07

list task

Display a list of tasks.

Synopsis

list task [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-start <Task start date>] [-end <Task end date>] [-parentid <Parent task ID>] [-status <Task status>] [-id <Task ID>] [-type <Task Type>]

Description

This command behaves differently depending on which options you use. The command by itself returns a list of all tasks. Each option filters the returned list of tasks, causing it to return a subset of the total list.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.: Display only those tasks associated with the specified device.
-host - A valid hostname: Display only those tasks associated with the specified device.
-fqdn - A valid Fully Qualified Domain Name: Display only those tasks associated with the specified device.
-deviceid - A valid device ID: Display only those tasks associated with the specified device.
-start - YYYY:MM:DD:HH:mm: Display only those tasks whose schedule date falls on or after the given date.
-end - YYYY:MM:DD:HH:mm: Display only those tasks whose schedule date falls on or before the given date
-parentid - a task ID: Display only those tasks whose parent is the task specified by the given Task ID.
-status - (pending | succeeded | failed | running | paused | starting | waiting | synchronous | skipped | warning): Display only those tasks with the specified status.
-id - a task ID: Display the task with the given task ID.
-type - Task type: Display the specific type of task. Possible values:Take SnapshotDiscover DriverRun ICMP TestDeploy PasswordsImportClean Devices after ImportDeploy ConfigResolve FQDNConfigure SyslogRun Command

ScriptRun DiagnosticsRun External ApplicationSynchronize Startup and RunningSend Email DigestClean Debug Log FilesUpdate Device SoftwareBackup Device SoftwareDownload Image from Cisco.comOS AnalysisEmail ReportRun Software Upload ScriptReboot DevicePrepare Device MemoryCheck Policy ComplianceGenerate Summary ReportsData PruningRun Device ScriptMulti-Task ProjectDelete ACLsDetect Network DevicesDeduplicationDeploy Remote Agent

Return Type

Collection:ScheduleTaskVO with columns:

approvalPriority
approvalStatus
approvalUsers
approveByDate
comments
completeDate
coreID
createDate
csvData
csvGroupTask
deviceDataID
deviceGroupID
deviceID
duration
expensive
failedChildCount
failureType
lastModifiedDate
parentTaskID
pendingChildCount
repeatCount
repeatEndDate
repeatInterval
repeatType
repeatWeekday
reservationEndDate
reservedDuration
reservedDurationType
result
resultConfigID
retryCount
retryInterval
runASAP
scheduleDate
scheduleTaskID
siteID
startDate
status
subTask

subTasks
succeededChildCount
systemTask
taskCustom1
taskCustom2
taskCustom3
taskCustom4
taskCustom5
taskCustom6
taskData
taskName
taskType
taskUserID
ticketNumber
userGeneratedChange

Examples

list task -parentid 78
list task -start 2004:02:29:23:59 -status failed
list task -start 2004:02:29:00:00 -end 2004:03:01:00:00 -ip 192.0.2.10
list task -id 23
list task -type "Take Snapshot"

list task all

List all tasks.

Synopsis

list task all

Description

Equivalent to "list task".

Return Type

Collection:ScheduleTaskVO with columns:

approvalPriority
approvalStatus
approvalUsers
approveByDate
comments
completeDate
coreID
createDate
csvData
csvGroupTask

deviceDataID
deviceGroupID
deviceID
duration
expensive
failedChildCount
failureType
lastModifiedDate
parentTaskID
pendingChildCount
repeatCount
repeatEndDate
repeatInterval
repeatType
repeatWeekday
reservationEndDate
reservedDuration
reservedDurationType
result
resultConfigID
retryCount
retryInterval
runASAP
scheduleDate
scheduleTaskID
siteID
startDate
status
subTask
subTasks
succeededChildCount
systemTask
taskCustom1
taskCustom2
taskCustom3
taskCustom4
taskCustom5
taskCustom6
taskData
taskName
taskType
taskUserID
ticketNumber
userGeneratedChange

Examples

list task all

list template devices

List template devices.

Synopsis

list template devices -id <Template ID>

Description

Lists all devices in the system which can use the given template

-id - List all those devices applicable to that template for the given template ID

Return Type

Collection:DeviceVO with columns:

accessMethods
assetTag
changeEventData
comments
consoleIPAddress
consolePort
consoleRealmName
contact
createDate
deviceCustom1
deviceCustom2
deviceCustom3
deviceCustom4
deviceCustom5
deviceCustom6
deviceGroup1ID
deviceGroup2ID
deviceGroup3ID
deviceID
deviceName
deviceType
driverName
duplexMismatchDetected
excludeFromPoll
feedSource
firmwareVersion
flashMemory
freePorts
geographicalLocation
hierarchyLayer
hostName
lastAccessAttemptDate
lastAccessAttemptStatus
lastAccessSuccessDate

lastConfigChangeUserID
lastConfigPolicyCheckedDate
lastDuplexDataUpdate
lastImportDate
lastModifiedUserID
lastRecordModifiedDate
lastSnapshotAttemptDate
lastSnapshotAttemptStatus
lastSnapshotDate
lastSnapshotSuccessDate
lastTopologyDataUpdate
lastUsedAuthenticationID
latestSoftwareHistoryID
latestStartupRunningDiffer
managementStatus
memory
model
modemNumber
mostRecentConfigID
nATIPAddress
nATRealmName
performACLParsing
policyImportance
policyInCompliance
primaryFQDN
primaryIPAddress
processor
putInServiceDate
rOMVersion
realmName
serialNumber
siteID
siteName
softwareVersion
tFTPServerIPAddress
ticketNumber
totalPorts
unmanagedDate
uptime
uptimeStoredDate
vendor

Examples

list template devices -id 601

list topology

List device information from topology data.

Synopsis

list topology [-mac <MAC Address>] [-ip <IP Address>]

Description

-mac - Show only devices that have seen this MAC address (no colons)
-ip - Show only devices that have seen this IP Address

Return Type

Collection:DeviceTopologyConnectionVO with columns:

crossReferenceTopologyDataID
data
deviceID
devicePortID
deviceTopologyDataID
firstSeen
ipAddress
lastSeen
lastUpdated
remoteDeviceID
remoteDevicePortID
remoteSasServerID
remoteSasServerInterfaceID
topologyDataType

Examples

list topology
list topology -mac 00AABBCCDDEE
list topology -ip 192.0.2.10
list topology -mac 00AABBCCDDEE -ip 192.0.2.10

list topology graph

List probable Layer 1 topology data.

Synopsis

list topology graph [-deviceids <List of Device IDs>] [-deviceportids <List of Device Port IDs>] [-serverids <List of Server IDs>] [-serverportids <List of Server Interface IDs>] [-deviceid <A Device ID>]

Description

-deviceids - A comma separated list of device IDs
-deviceportids - A comma separated list of device port IDs
-serverids - A comma separated list of server IDs
-serverportids - A comma separated list of server interface IDs
-deviceid - A device ID

Return Type

Collection:DeviceTopologyDataVO with columns:

crossReferenceTopologyDataID
data
deviceID
devicePortID
deviceTopologyDataID
firstSeen
lastSeen
lastUpdated
remoteDeviceID
remoteDevicePortID
remoteSasServerID
remoteSasServerInterfaceID
topologyDataType

Examples

list topology graph -deviceid 193
list topology graph -deviceids 54302,16001

list topology ip

List ip address topology data.

Synopsis

list topology ip [-deviceip <Device IP/hostname>] [-portid <Device Port ID>] [-notcurrent < >] [-type <(all|internal|connected)>] [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-deviceip - The ip/hostname of the device for which to show IP topology data.
-portid - The port id for which to show IP topology data.
-notcurrent - Specify to limit output to only IP topology data that is no longer visible.
-type - Limit the IP data to a specific type.
-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID

Return Type

Collection:DeviceTopologyDataVO with columns:

crossReferenceTopologyDataID
data
deviceID
devicePortID
deviceTopologyDataID
firstSeen
lastSeen
lastUpdated
remoteDeviceID
remoteDevicePortID
remoteSasServerID
remoteSasServerInterfaceID
topologyDataType

Examples

list topology ip -deviceip 192.0.2.10
list topology ip -portid 54302
list topology ip -deviceip 192.0.2.10 -notcurrent
list topology ip -portid 54302 -type internal
list topology ip -deviceid 1401

list topology mac

List MAC address topology data.

Synopsis

list topology mac [-deviceip <Device IP/hostname>] [-portid <Device Port ID>] [-notcurrent < >] [-type <(all|internal|connected)>] [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-deviceip - The ip/hostname of the device for which to show MAC topology data.
-portid - The port id for which to show MAC topology data.
-notcurrent - Specify to limit output to only MAC topology data that is no longer visible.
-type - Limit the MAC data to a specific type.
-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname

-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID

Return Type

Collection:DeviceTopologyDataVO with columns:

crossReferenceTopologyDataID
data
deviceID
devicePortID
deviceTopologyDataID
firstSeen
lastSeen
lastUpdated
remoteDeviceID
remoteDevicePortID
remoteSasServerID
remoteSasServerInterfaceID
topologyDataType

Examples

list topology mac -deviceip 192.0.2.10
list topology mac -portid 54302
list topology mac -deviceip 192.0.2.10 -notcurrent
list topology mac -portid 54302 -type internal
list topology mac -deviceid 1401

list trunk port

Lists all trunk ports on a specific device

Synopsis

list trunk port [-deviceip <Device IP address>] [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-deviceip - The device's ip address a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-host - A valid hostname

-fqdn - A valid Fully Qualified Domain Name

-deviceid - A device ID

Return Type

Collection:DevicePortVO with columns:

associatedChannelID
associatedChannelName
associatedVlanID
comments
configuredDuplex
configuredSpeed
description
deviceID
devicePortID
ipAddresses
lastModifiedDate
macAddress
nativeVlan
negotiatedDuplex
negotiatedSpeed
portAllows
portCustom1
portCustom2
portCustom3
portCustom4
portCustom5
portCustom6
portName
portState
portStatus
portType
slotNumber
temporaryVlanName

Examples

list trunk port -deviceid 201

list user

List all users known to the system.

Synopsis

list user

Return Type

Collection:UserVO with columns:

aaaPassword

aaaUserName
allowFailover
comments
createDate
deviceGroup1ID
deviceGroup2ID
deviceGroup3ID
distinguishedName
emailAddress
firstName
lastLoginDate
lastModifiedDate
lastName
privilegeLevel
requiredUser
status
ticketNumber
timeZone
useAaaLoginForProxy
userCustom1
userCustom2
userCustom3
userCustom4
userCustom5
userCustom6
userID
userName
userPassword

Examples

list user

list user id

List all users viewable by a particular user.

Synopsis

list user id [-viewable_by <Viewable By>]

Description

-viewable_by - List only users that are viewable by this user_ID

Return Type

Collection:IntegerVO

Examples

list user id -viewable_by 201

list view

Display the views defined within the system.

Synopsis

list view

Return Type

Collection:DeviceViewVO with columns:

m_appliesTo
m_defaultDeviceGroupID
m_description
m_enabled
m_lastModifiedDate
m_viewID
m_viewName
partitionCount
partitions

Examples

list view

list vlan

Show the vlans and their associated Device Port ID on a device.

Synopsis

list vlan [-deviceip <Device IP/Hostname>] [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-deviceip - The ip/hostname of the device for which to show vlans.
-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID

Return Type

Collection:DevicePortVO with columns:

associatedVlanID
comments
configuredDuplex
configuredSpeed
description
deviceID
devicePortID
ipAddresses
lastModifiedDate
macAddress
negotiatedDuplex
negotiatedSpeed
portAllows
portCustom1
portCustom2
portCustom3
portCustom4
portCustom5
portCustom6
portName
portState
portStatus
portType
slotNumber
temporaryVlanName

Examples

list vlan -deviceip 192.0.2.10
list vlan -deviceid 1401

list vlan on port

List Vlan's on a specific Port.

Synopsis

list vlan on port -portid <Port ID>

Description

List Vlan's on a specific Port.

-portid - The ID of a port

Return Type

Collection:PortVlanVO with columns:

deviceVlanInfoID

vlanID
vlanName
vlanPortType

Examples

list vlan on port -portid 521

list vlan ports

Show the ports on a given vlan identified by its port id.

Synopsis

list vlan ports -id <VLAN port id>

Description

-id - The port id of the vlan (provided in 'list vlan').

Return Type

Collection:DevicePortVO with columns:

associatedVlanID
comments
configuredDuplex
configuredSpeed
description
deviceID
devicePortID
ipAddresses
lastModifiedDate
macAddress
negotiatedDuplex
negotiatedSpeed
portAllows
portCustom1
portCustom2
portCustom3
portCustom4
portCustom5
portCustom6
portName
portState
portStatus
portType
slotNumber
temporaryVlanName

login

Synopsis

login -username <Username> -password <Password> [-host <Host>]

Description

-username - @ProductAbbreviation@ username
-password - @ProductAbbreviation@ password
-host - URL of @ProductAbbreviation@ server (defaults to localhost:@JNP_PORT@)

Return Type

None

logout

Synopsis

logout -sessionid <Session ID>

Description

-sessionid - @ProductAbbreviation@ SOAP API Session ID

Return Type

None

mod advanced script

Modify an existing advanced script.

Synopsis

mod advanced script [-id <Script ID>] [-name <Script Name>] [-newname <New Name>] [-description <New Description>] [-scripttype <New Script Type>] [-family <New Device Family>] [-language <New Script Language>] [-parameters <New Parameters>] [-script <New Script Text>] [-sitename <Site name>]

Description

Modify the indicated advanced script. Specify the script by ID or name. If more than one name match occurs, then an error will be reported and you must specify the unique script desired by ID.

-id - ID of the advanced script to edit
-name - Name of the advanced script to edit
-newname - New name for the script being modified

-description - New description for the script being modified
-scripttype - New script type (i.e. user defined subcategory)
-family - New device family for the script being modified
-language - New language for the script being modified - must be a supported language such as Expect or Perl
-parameters - New command line parameters for the script being modified
-script - New script text
-sitename - Site name

Return Type

Status

Examples

mod advanced script -id 22 -newname "Set Duplex" -description "Sets the interface duplex configuration" -scripttype "Interface Management Scripts" -sitename "Default Site"
mod advanced script -name "Extended Ping" -family "Cisco IOS" -language "Expect" -parameters "-l /usr/etc/log.txt" -script "send \"extended ping $Target_IP$\""

mod authentication

Modify device password information.

Synopsis

mod authentication -loc <Location> [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-snmpro <Read only community string(s)>] [-snmprw <Read write community string(s)>] [-snmpv3user <SNMPv3 Username>] [-snmpv3authpw <SNMPv3 Authentication Password>] [-snmpv3encryptpw <SNMPv3 Encryption Password>] [-user <Username>] [-passwd <Password>] [-enableuser <Enable username>] [-enablepasswd <Enable password>] [-connectionmethods <Connection methods>] [-accessvariables <Access variables>] [-start <Task start date>] [-appendsnmpro] [-appendsnmprw] [-sync] [-group <Group name>] [-site <Password Rule partition site>] [-rulename <Password Rule name>] [-rulehostname <Rule hostname>] [-ruledevicegroup <Rule device group>] [-iprangestart <Rule ip start range>] [-iprangeend <Rule ip end range>]

Description

This command can modify passwords on a specific device, across all devices in a device group, or update what the system knows of the device's password information. When using this command to modify passwords on a device or device group, the modification operation is actually a scheduled task.

-loc - The location to which password information should be written. Valid values for this argument are "db", "device", and "group". "db" tells the command that

password information should be changed only in the system's database. "device" tells the command that the password changes should be made on the device as well and "group" performs the same function as "device" but across all devices in the group.
-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.: An existing device to which this password information should apply.
-host - A valid hostname: An existing device to which this password information should apply.
-fqdn - A valid Fully Qualified Domain Name: An existing device to which this password information should apply.
-deviceid - A valid device ID: An existing device to which this password information should apply.
-snmpro - When used in conjunction with -loc db, this argument is taken as a single community string understood by the system as THE read only community string for the device or network. When used in conjunction with -loc device, this argument is taken as a comma-seperated list of read only community strings to be, either set on the device, or appended to an existing list of read only community strings (depends on whether or not the -appendsnmpro flag was supplied.)
-snmprw - When used in conjunction with -loc db, this argument is taken as a single community string understood by the system as THE read write community string for the device or network. When used in conjunction with -loc device, this argument is taken as a comma-seperated list of read write community strings to be, either set on the device, or appended to an existing list of read write community strings (depends on whether or not the -appendsnmprw flag was supplied.)
-snmpv3user - When used in conjunction with -loc db, this argument is taken as the username for snmpv3 access.
-snmpv3authpw - When used in conjunction with -loc db, this argument is taken as the authentication password for snmpv3 access.
-snmpv3encryptpw - When used in conjunction with -loc db, this argument is taken as the encryption password for snmpv3 access.
-user - Username.
-passwd - Password.
-enableuser - ADDITIONAL username to get to "enable" mode.
-enablepasswd - ADDITIONAL password to get to "enable" mode.
-connectionmethods - The methods used by the system to connect to devices. Can be telnet, serial_direct, or SSH.
-accessvariables - To override variables in the script, such as prompts.
-start - YYYY:MM:DD:HH:mm. The first date on which the task will run. Use this option only if the argument to the -loc flag is "device".
-appendsnmpro - Supply this option if read only community strings should be appended to any existing on the device. Use this option only if the argument to the -loc flag is "device".
-appendsnmprw - Supply this option if read write community strings should be appended to any existing on the device. Use this option only if the argument to the -loc flag is "device".
-sync - Indicates that the command should return only after the password change task is complete. Do not use this option with -start.

-group - The group name for performing this command across all devices in a group.
-site - The site partition this rule belongs to. Default to be global
-rulename - The password rule name to apply the access variables to
-rulehostname - Hostname, the rule applies to
-ruledevicegroup - Device group name, the rule applies to
-iprangestart - IP range start (range), the rule applies to
-iprangeend - IP range end (range), the rule applies to

Return Type

String

Examples

mod authentication -loc db -ip 192.0.2.10 -passwd fish -snmpro public -enablepasswd 31337
mod authentication -loc device -ip 192.0.2.10 -passwd limited -enablepasswd full
mod authentication -loc device -ip 192.0.2.10 -passwd some -enablepasswd all -snmprw brillig,slithy,toves,gire -appendsnmprw -sync
mod authentication -loc device -ip 192.0.2.10 -passwd less -enablepasswd more -snmpro foo,bar,fork,snork -start 2004:02:29:23:59
mod authentication -loc group -group MyDevices -passwd less -enablepasswd more -snmpro foo,bar,fork,snork -start 2004:02:29:23:59
mod authentication -loc db -rulename "rule 1" -rulehostname DALAB-C2600-NAT
mod authentication -loc db -rulename "rule 2" -ruledevicegroup DeviecGroup1
mod authentication -loc db -site DefaultSite -rulename "rule 3" -iprangestart 172.30.1.1 -iprangeend 172.30.1.5

mod command script

Modify an existing command script.

Synopsis

mod command script [-id <Script ID>] [-name <Script Name>] [-newname <New Name>] [-description <New Description>] [-scripttype <New Script Type>] [-mode <New Mode>] [-driver <New Driver List>] [-script <New Script Text>] [-sitename <Site Name>]

Description

Modify the indicated command script. The desired script can be specified by ID or name. If more than one name match occurs, then an error will be reported and you must specify the unique script desired by ID.

-id - ID of the command script to edit
-name - Name of the command script to edit
-newname - New name for the script being modified
-description - New description for the script being modified

-scripttype - New script type (i.e. user defined subcategory)
-mode - New command script mode
-driver - New list of applicable drivers - provided as a comma separated list of internal driver names
-script - New script text
-sitename - Site Name

Return Type

Status

Examples

mod command script -id 22 -newname "Set Duplex" -description "Sets the interface duplex configuration" -scripttype "Interface Management Scripts" -sitename "Default Site"
mod command script -name "Extended Ping" -mode "Cisco IOS enable" -driver "CiscoIOSGeneric,CiscoIOSSwitch" -script "extended ping $Target_IP$"

mod custom data

Modify an existing piece of custom data.

Synopsis

mod custom data -tablename <Table Name> -columnname <Column Name> [-fieldname <Field Name>] [-fieldlabel <Field Label>] [-flags <Flags>] [-inuse <In Use>] [-fieldvalues <Field Values>]

Description

-tablename - The custom data used for the table
-columnname - Column name in the table
-fieldname - Field name of the custom data
-fieldlabel - Field label of the custom data
-flags - The flags of the custom data
-inuse - The in use status of the custom data
-fieldvalues - The field values for the custom data

Return Type

CustomDataVO with columns:

columnName
customDataID
fieldLabel
fieldName
fieldValues
flags
inUse

lastModifiedDate
tableName

Examples

mod custom data -tablename RN_DEVICE -columnname DeviceCustom1 -fieldname UUID -fieldlabel NNMUuid -flags 1 -inuse 1
mod custom data -tablename RN_DEVICE_GROUP -columnname DeviceGroupCustom5 -fieldname Test -fieldlabel Test1 -flags 0 -inuse 1

mod device

Modify the properties of a device.

Synopsis

mod device [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-hostname <New Hostname>] [-comment <Comment>] [-description <Device name>] [-model <Device model>] [-vendor <Device vendor>] [-domain <Domain name>] [-serial <Serial number>] [-asset <Asset tag>] [-location <Location>] [-unmanaged <Unmanaged>] [-nopoll <Do not poll>] [-newIP <New IP address>] [-consoleip <Console IP address, if using console server>] [-consoleport <Console Port>] [-tftpserverip <TFTP server IP address, if using NAT>] [-natip <NAT IP address>] [-customname <Customname>] [-customvalue <Customvalue>] [-useconsoleserver <true or false>] [-accessmethods <Comma-separated list of access methods>] [-hierarchylayer <Hierarchy layer>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-hostname - The device's new host name
-comment - Additional information regarding the device.
-description - The descriptive name of the device (informational only).
-model - The device's model (such as 2620).
-vendor - The device's vendor (such as Cisco).
-domain - A fully qualified domain name (such as www.google.com).
-serial - The device's serial number.
-asset - The device's asset tag.
-location - The device's location.
-unmanaged - 0: Mark this device as managed by the system. 1: Mark this device to be unmanaged by the system.
-nopoll - 0: Mark this device to be polled for changes. 1: Mark this device as not to be polled for changes.

-newIP - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device will be put in. This is the new IP address of the device.
-consoleip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with REALM_NAME:, where REALM_NAME is the name of the Realm the address is in.
-consoleport - The port number
-tftpserverip - a.b.c.d where 0 <= a,b,c,d <= 255
-natip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with REALM_NAME:, where REALM_NAME is the name of the Realm the address is in.
-customname - The custom field name
-customvalue - The custom field value
-useconsoleserver - true, if the device uses a console server. false, if the device does not.
-accessmethods - A comma-separated list of access methods, or "none". The set of access methods: {telnet, ssh, rlogin, SCP, FTP, TFTP, SNMP, snmp_noauthnopriv, snmp_authnopriv, snmp_authpriv}.
-hierarchylayer - This device attribute is used in diagramming. When you config a network diagram, you can select which hierarchy layers on which to filter. Valid values include: (core, distribution, access, edge and "layer not set").

Return Type

Status

Examples

mod device -ip 192.0.2.10 -newIP 192.0.2.10
mod device -ip 192.0.2.10 -newIP "West Site:192.0.2.10"
mod device -ip "East Site:192.0.2.10" -newIP "West Site:192.0.2.10"
mod device -ip 192.0.2.10 -nopoll 1 -comment "enabled polling by change detection."
mod device -ip 192.0.2.10 -customname Owner -customvalue Bob
mod device -ip 192.0.2.10 -useconsolesever false

mod device relationship

Modify the relationship type defined between the devices.

Synopsis

mod device relationship -relationshipid <Relationship ID> -newrelationshiptypeid <New Relationship Type ID>

Description

Modify the relationship identified by relationship id between two devices to a new relationship type.

-relationshipid - Specify the Relationship ID for the relationship which you want to modify.

-newrelationshiptypeid - Specify the new Relationship type ID, 1: PC_USER , 2:PC_GENERIC , 3:PC_ADMIN_CONTEXT, 4:PEER_USER, 5:PEER_GENERIC

Return Type

VO:DeviceRelationshipVO with columns:

createdBy
destinationDeviceID
deviceRelationshipID
lastModifiedDate
originatingDeviceID
relationshipTypeID

Examples

mod device relationship -relationshipid 101 -newrelationshiptypeid 3

mod device template

Modify the properties of a device template.

Synopsis

mod device template -templateid <Device Template ID> [-hostname <Device name>] [-newdriver <Driver name>] [-comment <Comment>] [-description <Description>] [-model <Device model>] [-vendor <Device vendor>] [-location <Location>] [-customname <Customname>] [-customvalue <Customvalue>] [-accessmethods <Comma-separated list of access methods>] [-hierarchylayer <Hierarchy layer>] [-sitename <Site Name>]

Description

-templateid - A device template ID

-hostname - A valid name

-newdriver - The new device driver name in short form

-comment - Additional information regarding the device template.

-description - The descriptive name of the device template (informational only).

-model - The device template's model (such as 2620).

-vendor - The device's vendor (such as Cisco).

-location - The device's location.

-customname - The custom field name

-customvalue - The custom field value

-accessmethods - A comma-separated list of access methods, or "none". The set of access methods: {telnet, ssh, rlogin, SCP, FTP, TFTP, SNMP,snmp_noauthnopriv, snmp_authnopriv_sha, snmp_authnopriv_md5,

snmp_authpriv_sha_des, snmp_authpriv_sha_aes, snmp_authpriv_sha_aes192, snmp_authpriv_sha_aes256, snmp_authpriv_md5_des, snmp_authpriv_md5_aes, snmp_authpriv_md5_aes192, snmp_authpriv_md5_aes256}.

-hierarchylayer - This device attribute is used in diagramming. When you config a network diagram, you can select which hierarchy layers on which to filter. Valid values include: (core, distribution, access, edge and "layer not set").

-sitename - The Site name in which the template belongs to.

Return Type

STATUS

Examples

mod device template -templateid 801 -comment "Test Comment"

mod device template -templateid 801 -customname Owner -customvalue Bob

mod device template -templateid 801 -accessmethods FTP,SSH

mod device template config

Modify the config of a device template.

Synopsis

mod device template config -templateid <Device Template ID> [-configtext <Device template configuration text>] [-configfile <Device template config file name>]

Description

-templateid - A device template ID

-configtext - A valid configuration text in double quotes

-configfile - Specify the absolute path to the file which contains the device template configuration. The file must be directly accessible by the system.

Return Type

STATUS

Examples

mod device template config -templateid 801 -configtext "$var1"

mod device template config -templateid 801 -configile /usr/home/config.txt

mod diagnostic

Modify an existing custom diagnostic script.

Synopsis

mod diagnostic [-id <Diagnostic ID>] [-name <Diagnostic Name>] [-newname <New Name>] [-description <New Description>] [-mode <New Mode>] [-driver <New Driver List>] [-script <New Script Text>] [-sitename <Site name>]

Description

Modify the indicated diagnostic script. The desired diagnostic can be specified by ID or name. If more than one name match occurs, then an error will be reported and you must specify the unique diagnostic desired by ID.

-id - ID of the diagnostic to edit
-name - Name of the diagnostic to edit
-newname - New name for the diagnostic being modified
-description - New description for the diagnostic being modified
-mode - New command script mode
-driver - New list of applicable drivers - provided as a comma separated list of internal driver names
-script - New diagnostic script text
-sitename - Site name

Return Type

Status

Examples

mod diagnostic -id 22 -newname "Show IP CEF" -description "Gather IP CEF information" -sitename "Default Site"
mod diagnostic -name "Extended Ping To Core" -mode "Cisco IOS enable" -driver "CiscoIOSGeneric,CiscoIOSSwitch" -script "extended ping 192.0.2.10"

mod group

Modify the comments associated with and/or the name of a group.

Synopsis

mod group -type <Type> -name <Name> [-newname <New name>] [-comment <Comment>] [-customname <Customname>] [-customvalue <Customvalue>] [-shared <Shared>]

Description

-type - The type of the group. "device" is currently the only valid argument to this option.
-name - The name of the group to be modified.

-newname - The new name for the modified group. Do not use this option unless you also use -name.
-comment - Additional information regarding the group.
-customname - The custom field name
-customvalue - The custom field value
-shared - 1 if the group is shared, 0 if it is not.

Return Type

Status

Examples

mod group -name "mystery routers" -type device -comment "removing these devices is a bad idea, but we don't really know what purpose they serve."
mod group -type device -name "border routers" -newname "defunct"
mod group -type device -name "border routers" -customname Location -customvalue Earth

mod ip

Modify the properties of an ip address.

Synopsis

mod ip -ipvalue <Value> [-deviceip <Device IP address>] [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-comment <Comment>] [-usetoaccess <Use to Access Device>]

Description

-ipvalue - The ip value a.b.c.d where 0 <= a,b,c,d <= 255
-deviceip - The device's ip address a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-comment - Additional information regarding the device.
-usetoaccess - Use this IP Value to access its device, 0 - yes, 1 - no, default - no

Return Type

String

Examples

mod ip -deviceip 192.0.2.10 -ipvalue 192.0.2.10 -comment "my own ip"
mod ip -deviceip 192.0.2.10 -ipvalue 192.0.2.10 -usetoaccess 0

mod ip -deviceid 1401 -ipvalue 192.0.2.10 -usetoaccess 0

mod metadata

Modify an existing piece of custom data associated with a specific field and associated entity.

Synopsis

mod metadata -metadataid <Metadata ID> [-fieldid <Metadata Field ID>] [-data <Data>] [-associatedtableid <Matching Row ID>]

Description

-metadataid - ID of the custom data to delete
-fieldid - Field ID the data is to be associated with
-data - New data to be associated
-associatedtableid - ID of the associated row the data coresponds to

Return Type

MetadataVO with columns:

associatedTableID
data
fieldID
lastModifiedDate
metadataID

Examples

mod metadata -metadataid 99381 -fieldid 121 -data Room102
mod metadata -metadataid 99381 -data Room101 -associatedtableid 21031

mod metadata field

Modify an existing custom data field.

Synopsis

mod metadata field -fieldid <Field ID> [-fieldname <Field Name>] [-fieldvalues <Field Values>] [-inuse <In Use>] [-flags <Allow HTML>] [-associatedtable <Associated Table>]

Description

-fieldid - ID of the field to modify
-fieldname - New name of the field

-fieldvalues - List of comma separated values that the field is restricted to. If not specified, the value for this field is not restricted
-inuse - Turns the field on or off. 1 is on, 0 is off. When the field is off, it will not be displayed with the other custom fields.
-flags - Used for allowing HTML in the field value. 1 is allow, 0 is disallow. If disallowed, HTML will be escaped for displaying.
-associatedtable - The table to associate this field with

Return Type

MetadataFieldVO with columns:

associatedTable
fieldDataType
fieldID
fieldName
fieldValues
flags
inUse
lastModifiedDate

Examples

mod metadata field -fieldid 2221 -fieldname Room -fieldvalues 101,102,103,104 -inuse 1 -flags 0 -associatedtable RN_DEVICE
mod metadata field -fieldid 2221 -inuse 0

mod module

Modify a module's properties.

Synopsis

mod module -id <Module ID> [-comment <Comment>] [-customname <Customname>] [-customvalue <Customvalue>]

Description

-id - The ID of a module
-comment - Additional information about the module.
-customname - The custom field name
-customvalue - The custom field value

Return Type

Status

Examples

mod module -id 527 -comment "Internal Use Only"

mod partition

Modify a partition.

Synopsis

mod partition -name <Name> -newname <New name> [-comment <Comment>]

Description

-name - The name of the partition to be modified.
-newname - The new name for the modified partition. Do not use this option unless you also use -name.
-comment - Additional information regarding the partition.

Return Type

Status

Examples

mod partition -name "Default Site" -newname "Redmond Site"

mod port

Modify a port's properties.

Synopsis

mod port -id <Port ID> [-comment <Comment>] [-customname <Customname>] [-customvalue <Customvalue>]

Description

-id - The ID of a port
-comment - Additional information about the port.
-customname - The custom field name
-customvalue - The custom field value

Return Type

Status

Examples

mod port -id 527 -comment "Internal Use Only"

mod task

Modify a scheduled task.

Synopsis

mod task -id <Task ID> [-comment <Comment>] [-retryInterval <Retry interval>] [-expensive] [-notexpensive] [-days <Days>] [-retryCount <Retry count>] [-repeatType <Repeat type>] [-duration <Duration>] [-start <Start>] [-repeatInterval <Repeat interval>] [-approve <Approval comment>] [-reject <Reason the task is not approved>] [-override <Reason for overriding approval process>] [-customname <Custom name>] [-customvalue <Custom value>] [-sessionlog <true or false>]

Description

-id - The task ID of the task to modify.
-comment - Additional information about the task.
-retryInterval - The number of seconds between retries.
-expensive - Mark the task as expensive. Do not use this option with -notexpensive.
-notexpensive - Mark the task as not expensive. Do not use this option with -expensive.
-days - This argument differs depending on the task.For weekly tasks, -days should be a comma-separated list of weekdays. Each item in the list is a day of the week upon which the task should be run. Valid weekdays are: sun, mon, tue, wed, thur, fri, sat.For monthly tasks, -days should be a single integer between 1 and 31, corresponding to the day of the month upon which the task should be run.
-retryCount - The number of times to retry the task if it fails.
-repeatType - The metric by which a task repeats. Valid values are 1: once, 2: periodically, 3: daily, 4: weekly, 5: monthly. If you modify this value, then modify -repeatInterval or -days accordingly.
-duration - Estimated duration the task will run(in minutes)
-start - YYYY:MM:DD:HH:mm. The first date the task will run. The string "now" means the current time. The string "tomorrow" means 24 hours from the current time.
-repeatInterval - This option differs depending on the task.For Periodic tasks, this is the period in minutes.For Monthly tasks, each bit of the integer (except the last) represents a day, but we recommend using the -days option to modify the days on which a monthly task runs.This option is invalid with all other tasks.
-approve - Approve the task
-reject - Reject the task
-override - Override the approval requirement
-customname - The custom field name
-customvalue - The custom field value
-sessionlog - If true a complete session log will be saved with this task.

Return Type

Status

Examples

mod task -id 7097 -repeatType 4 -days mon,wed,thur

mod task -id 7097 -start now
mod task -id 54 -retryCount 2 -duration 60
mod task -id 54 -reject "needs technical review"

mod topology graph

Modify topology data.

Synopsis

mod topology graph -type <Topology data type> -data <Topology data value> -deviceid <Device ID> [-deviceportid <Device ID>] [-remotedeviceid <Device ID>] [-remotedeviceportid <Device ID>] [-serverid <Server ID>] [-serverportid <Server ID>]

Description

-type - The topology data type, typically "phy_inferred" for L1
-data - The topology data value, typically a MAC address (without colons)
-deviceid - The source device ID
-deviceportid - The source device port ID
-remotedeviceid - The destination device ID
-remotedeviceportid - The destination device port ID
-serverid - The destination server ID
-serverportid - The destination server port ID

Return Type

String

Examples

mod topology graph -type phy_inferred -data 0007E912C8D7 -deviceid 193 -remotedeviceid 2837
mod topology graph -type phy_inferred -data 00123F76F759 -deviceid 193 -serverid 105001

mod unmanaged device

Modify the properties of an unmanaged device.

Synopsis

mod unmanaged device -ip <IP address> -comment <Comment>

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-comment - Additional information regarding the device.

Return Type

Status

Examples

mod unmanaged device -ip 192.0.2.10 -comment "no need"

mod user

Modify a user's properties.

Synopsis

mod user -u <Username> [-p <Password>] [-fn <First name>] [-ln <Last name>] [-email <Email address>] [-priv <User Privilege>] [-newusername <Username>] [-aaausername <Username>] [-aaapassword <AAA Password>] [-useaaaloginforproxy <Use AAA Logins for Proxy (yes|no)>] [-extauthfailover <Allow External Auth Failover (yes|no)>] [-customname <Customname>] [-customvalue <Customvalue>] [-status <Enable or Disable the user (enable|disable)>] [-view1partition <view1partitionname>] [-view2partition <view2partitionname>] [-view3partition <view3partitionname>]

Description

-u - Username
-p - Password
-fn - First name
-ln - Last name
-email - Email address
-priv - User Privilege (1=Limited Access,2=Full Access,3=Power User,4=Admin)
-newusername - New username for this user.
-aaausername - AAA username for this user.
-aaapassword - AAA password for this user.
-useaaaloginforproxy - Whether to user AAA logins for the Proxy Interface for this user (yes|no).
-extauthfailover - Whether to allow external auth failover for this user (yes|no).
-customname - The custom field name
-customvalue - The custom field value
-status - enable or disable
-view1partition - partition the user belongs to in the first view
-view2partition - partition the user belongs to in the second view
-view3partition - partition the user belongs to in the third view

Return Type

Status

Examples

mod user -u johnd -p new -fn Johnathan -email jdoe@example.net

mod user -u johnd -p new -fn Johnathan -email jdoe@example.net -priv 2
mod user -u -customname Title -customValue Engineer
mod user -u johnd -status disable
mod user -u johnd -view1partition "Default Site" -view2partition Fedex -view3partition Security

mod user group

Modify a user group.

Synopsis

mod user group -name <Name> [-newname <New name>] [-description <Description>] [-cmdperms <command permissions>] [-deviceperms <device permissions>] [-scriptperms <script permissions>] [-viewperms <view permissions>] [-view1partition <view1parition name>] [-view2partition <view2parition name>] [-view3partition <view3parition name>]

Description

Modify the description of a usergroup.

-name - The name of the group to be modified.
-newname - The new name for the modified group.
-description - Description of the group.
-cmdperms - set command permission role(s) for the group
-deviceperms - set device permission role(s) for the group
-scriptperms - set script permission role(s) for the group
-viewperms - set view permission role(s) for the group
-view1partition - view1parition name
-view2partition - view2parition name
-view3partition - view3parition name

Return Type

Status

Examples

mod user group -name "Test User Group" -description "User group created for testing" -newname TestGroup1
mod user group -name "TestGroup" -cmdperms cmdpermission1 -deviceperms mdp1 -scriptperms scriptpermission1 -viewperms viewpermission1
mod user group -name "TestGroup" -view1partition "Default Site" -view2partition Fedex -view3partition Security

mod vlan

Edits a Vlan - Rename Vlan Name, Add ports or Remove ports

Synopsis

mod vlan [-deviceip <Device IP address>] [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] -vlanid <Vlan ID> [-renameto <New Vlan Name>] [-addports <Add Port IDs>] [-removeports <Remove Port IDs>] [-start <Task start date>] [-rep <Task repeat period>] [-sync] [-sessionlog <true or false>] [-retrycount <Retry count>] [-retryinterval <Retry interval>] [-comment <Snapshop comment>] [-presnapshot <true or false>] [-postsnapshot <true, false or task>] [-priority <Task priority>]

Description

-deviceip - The device's ip address a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-host - A valid hostname

-fqdn - A valid Fully Qualified Domain Name

-deviceid - A device ID

-vlanid - Vlan ID to edit

-renameto - New Vlan name

-addports - Ports that need to be added to the Vlan

-removeports - Ports that need to be removed from the Vlan

-start - YYYY:MM:DD:HH:mm. The first date on which the task will run.

-rep - (#min | #:# | #days | #weeks | #months) where # is a positive integer. #:# is hours:minutes--the two integers do not have to be the same. Do not use this option with -sync.

-sync - Indicates the command should return only after the snapshot retrieval task is complete. Do not use this option with -rep or -start.

-sessionlog - If true a complete session log will be saved with this task.

-retrycount - The number of times to retry the task if it fails.

-retryinterval - The number of seconds between retries.

-comment - An optional comment about the snapshot.

-presnapshot - If false, this indicates that the snapshot that runs before the script should be skipped.

-postsnapshot - If false, this indicates that the snapshot that runs after the script should be skipped. If "task", this indicates that snapshot after the script should run as a separate task.

-priority - Task priority value (1, 2, 3, 4 or 5). Invalid priority will be changed to an appropriate value automatically.

Return Type

String

Examples

mod vlan -deviceid 2801 -vlanid 109 -renameto VLAN_109 -addports 12,13,14 -removeports 18

mod vlan trunk

Edit a Vlan Trunk - Adds and Removes Vlans

Synopsis

mod vlan trunk [-deviceip <Device IP address>] [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] -portname <Port Name> -nativevlanid <Native vlan ID> [-addvlanids <Add Vlan IDs>] [-removevlanids <Remove Vlan IDs>] [-start <Task start date>] [-rep <Task repeat period>] [-sync] [-sessionlog <true or false>] [-retrycount <Retry count>] [-retryinterval <Retry interval>] [-comment <Snapshop comment>] [-presnapshot <true or false>] [-postsnapshot <true, false or task>] [-priority <Task priority>]

Description

-deviceip - The device's ip address a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-host - A valid hostname

-fqdn - A valid Fully Qualified Domain Name

-deviceid - A device ID

-portname - trunk port name to edit

-nativevlanid - specify a native or default vlan id

-addvlanids - vlan ids to add to trunk

-removevlanids - vlan ids to remove from trunk

-start - YYYY:MM:DD:HH:mm. The first date on which the task will run.

-rep - (#min | #:# | #days | #weeks | #months) where # is a positive integer. #:# is hours:minutes--the two integers do not have to be the same. Do not use this option with -sync.

-sync - Indicates the command should return only after the snapshot retrieval task is complete. Do not use this option with -rep or -start.

-sessionlog - If true a complete session log will be saved with this task.

-retrycount - The number of times to retry the task if it fails.

-retryinterval - The number of seconds between retries.

-comment - An optional comment about the snapshot.

-presnapshot - If false, this indicates that the snapshot that runs before the script should be skipped.

-postsnapshot - If false, this indicates that the snapshot that runs after the script should be skipped. If "task", this indicates that snapshot after the script should run as a separate task.

-priority - Task priority value (1, 2, 3, 4 or 5). Invalid priority will be changed to an appropriate value automatically.

Return Type

String

Examples

mod vlan trunk -deviceid 2801 -portname xxx -nativevlanid 1 -addvlanids 12,13,14 -removevlanids 21

os ping

Run a ping command from the server to the specified device.

Synopsis

os ping

Description

The ping command is an OS command. All ping options that are available at the OS level are supported. Users should be able to enter any host name or address. The behavior is that it simply passes the string to the OS, executes it as a command and returns the results of the executed command.

Return Type

None

Examples

os ping 192.0.2.10

os traceroute

Run a traceroute command from the server to the specified device.

Synopsis

os traceroute

Description

The traceroute command is an OS command. All traceroute options that are available at the OS level are supported. Users should be able to enter any options the command supported. The behavior is that it simply passes the command to the OS, executes it and returns the results of the executed command.

Return Type

None

Examples

os traceroute hp.com

os-ping

Run a ping command from the server to the sepecified device.

Synopsis

os-ping

Description

The ping command is an OS command. All ping options that are available at the OS level are supported. Users should be able to enter any host name or address. The behavior is that it simply passes the string to the OS, executes it as a command and returns the results of the executed command.

Return Type

None

Examples

os-ping 192.0.2.10

os-ping -t 192.0.2.11

passwd

Change current user's password.

Synopsis

passwd -oldpwd <your old password> -newpwd <your new password>

Description

Causes the current user's password to be changed.

-oldpwd - youroldpassword
-newpwd - yournewpassword

Return Type

Status

Examples

passwd -oldpwd youroldpassword -newpwd yournewpwd

pause polling

Stop polling.

Synopsis

pause polling

Description

Stop polling devices for configuration changes.

Return Type

Status

Examples

pause polling

ping

Run a ping command on a device.

Synopsis

ping -source <IP address | Hostname | Fully Qualified Domain Name> -sourcegroup <Groupname> -dest <List of IP addresses> -rep <Task repeat period> -async -start <task start date>

Description

Causes a series of ping commands to be executed on a device. One ping command is executed for each target host specified. This series of commands may by run on the device immediately, or scheduled to run sometime in the future. Via this command, the task scheduled can be set to repeat periodically. Note that if not scheduled as a task, this command may take some time to complete.

-source - Can be an IP address (a.b.c.d where 0 <= a,b,c,d <= 255), or a valid hostname, or a valid Fully Qualified Domain Name.
-sourcegroup - A valid group name. Exactly one of -source or -sourcegroup must be specified.

-dest - A comma seperated list of devices. Devices may be specified in any way that is understood by the ping program on the device specified by the option "-source".
-rep - (#min | #:# | #days | #weeks | #months) where # is a positive integer. #:# is hours:minutes, the two integers don't have to be the same. This option should not be used unless -async is also supplied.
-async - Indicates that the ping operation should be scheduled on the system as a task. The start time for the task will be immediatly unless an alternate start data is provided by means of the -start option.
-start - YYYY:MM:DD:HH:mm. The date on which the task will first be run. This option should not be used unless -async is also supplied.

Return Type

String

Examples

ping -source 192.0.2.10 -dest 192.0.2.10
ping -source 192.0.2.10 -dest 192.0.2.10,192.0.2.10,192.0.2.10
ping -source 192.0.2.10 -dest 192.0.2.10 -async -start 2004:02:29:23:59 -rep 2days
ping -source 192.0.2.10 -dest 192.0.2.10 -async
ping -sourcegroup mygroup -dest 192.0.2.10

provision device

Schedule a provision device task

Synopsis

provision device -ip <IP address> -templateid <Device Template ID> -priority <Task priority> [-name <A used defined name for the task>] [-start <Task start date>] [-variables <Variable List>] [-ignorevariables] [-comment <Comment>] [-duration <Duration>] [-sessionlog <true or false>] [-presnapshot <true or false>] [-postsnapshot <true, false or task>] [-retryCount <Retry count>] [-retryInterval <Retry interval>] [-nocompliance] [-setactive] [-copydata] [-rep <Task repeat period>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-templateid - Device template ID to provision the device.

-priority - Task priority value (1, 2, 3, 4 or 5). Invalid priority will be changed to an appropriate value automatically.

-name - A User defined name for indetyfying the task.

-start - YYYY:MM:DD:HH:mm. The first date the task will run. The string "now" means the current time. The string "tomorrow" means 24 hours from the current time.

-variables - A list of variables to be replaced in the script - provided as a list of name=value pairs, separated by commas. Values can be surrounded in single-quotes ('). Within a quoted value, a single-quote can be embedded with two single-quote characters. Example: "variable1=value1,varable2='this is ''value 2'''"

-ignorevariables - Mark the config variables as ignored that are not passed in the variables argument to the command.

-comment - Additional information about the provision device task.

-duration - Estimated duration the task will run(in minutes)

-sessionlog - If true a complete session log will be saved with this task.

-presnapshot - If false, this indicates that the snapshot that runs before the script should be skipped.

-postsnapshot - If false, this indicates that the snapshot that runs after the script should be skipped. If "task", this indicates that snapshot after the script should run as a separate task.

-retryCount - The number of times to retry the task if it fails.

-retryInterval - The number of seconds between retries.

-nocompliance - If passed policy compliance will not be checked before provisioning.

-setactive - Set device as active upon success.

-copydata - Copy additional information from device template to device.

-rep - (#min | #:# | #days | #weeks | #months) where # is a positive integer. #:# is hours:minutes--the two integers do not have to be the same. Do not use this option with -sync.

Return Type

String

Examples

provision device -ip 10.255.0.198 -templateid 1711 -start 2009:12:29:23:59 -setactive -copydata -nocompliance -priority 2 -sessionlog true -ignorevariables -variables "banner_string='Welcome to CiscoWorks NCM' -comment "UnitTesting" -rep 2weeks -presnapshot false -postsnapshot task -duration 60

quit

Exit the system.

Synopsis

quit

Description

Exit the system.

Return Type

None

Examples

quit

reload content

Load any new content packs.

Synopsis

reload content

Description

Load any new content packs (such as scripts or policies) that have been installed on the server since the last time it was restarted or content was reloaded.

Return Type

Status

Examples

reload content

reload drivers

Reload all installed driver files.

Synopsis

reload drivers -force

Description

Causes the server to reload all installed driver files.

-force - Force drivers to be reloaded even when there is error

Return Type

Status

Examples

reload drivers
reload drivers -force

reload plugins

Reload any code plugins.

Synopsis

reload plugins

Description

Reload any code plugins (such as the topology plugin) that may have been updated while the server was running.

Return Type

Status

Examples

reload plugins

reload server options

Reload config variables from all config files.

Synopsis

reload server options

Description

Causes the server to reload config variables from all config files.

Return Type

String

Examples

reload server options

resume polling

Resume polling.

Synopsis

resume polling

Description

Resume polling devices for configuration changes.

Return Type

Status

Examples

resume polling

rlogin

Make an rlogin connection to a device.

Synopsis

rlogin [-override] []

Description

Connect to a device through the system's Proxy Interface via telnet (bypassing single sign-on). If you are connected to a device through a console server, you may hit ctrl-\ to return to the system shell after logging out of the device.

-override - Force a connection to a device in the event that simultaneous connection warning or prevention is turned on.
- Hostname, Device ID, Fully Qualified Domain Name, or Primary IP Address to use to lookup the device to connect to. The characters * and ? can be used as wildcards.
- Port to use to connect to devices outside of the system.

Return Type

None

Examples

rlogin 192.0.2.10
rlogin -override mydevice

run advanced script

Run an existing advanced script against a device or group of devices.

Synopsis

run advanced script [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-group <Groupname>] -name <Script Name> [-parameters <Parameters>] [-variables <Variable List>] [-start <Task start date>] [-rep <Task repeat period>] [-sync] [-nowait] [-comment <Snapshop comment>] [-presnapshot <true or false>] [-postsnapshot <true, false or task>]

Description

Runs an existing advanced script, specified by name, against a device or group of devices. The proper variant of the script will be applied to each device. If no variant of the script supports a given device, that device will be skipped. The script is run as a system task.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-group - A valid group name. Either a device or a group must be specified, but not both (exactly one of -ip, -hostname, -fqdn or -group must be specified).
-name - Name of the advanced script to run
-parameters - Command line parameters for the advanced script to run
-variables - A list of variables to be replaced in the script - provided as a list of name=value pairs, separated by commas. Values can be surrounded in single-quotes ('). Within a quoted value, a single-quote can be embedded with two single-quote characters. Example: "variable1=value1,varable2='this is ''value 2'''"
-start - YYYY:MM:DD:HH:mm. The first date on which the task will run.
-rep - (#min | #:# | #days | #weeks | #months) where # is a positive integer. #:# is hours:minutes--the two integers do not have to be the same. Do not use this option with -sync.
-sync - Indicates the command should return only after the snapshot retrieval task is complete. Do not use this option with -rep or -start.
-nowait - Indicates that the task should not wait if there is another task currently running against the same device.
-comment - An optional comment about the snapshot.
-presnapshot - If false, this indicates that the snapshot that runs before the script should be skipped.
-postsnapshot - If false, this indicates that the snapshot that runs after the script should be skipped. If "task", this indicates that snapshot after the script should run as a separate task.

Return Type

String

Examples

run advanced script -ip 192.0.2.10 -name "Extended Ping" -parameters "" -variables "Target_IP=192.0.2.10" -start 2004:02:29:23:59 -rep 2days -comment "running extended ping"
run advanced script -group mygroup -name "Set Interface Description" -variables="interface=Ethernet1,description='provider ''MCI'',link id T207'" -parameters "-l /usr/etc/log.txt" -sync

run checkdb

Check the database for potential conflicts and optionally fix them.

Synopsis

run checkdb [-verbose] [-fix]

Description

The scan for conflicts will consist of known field issuses and their resolutions. It is not all-inclusive.

-verbose - Display verbose output, including SQL that would be used to fix the problems.
-fix - Flag to fix discovered problems

Return Type

Status

Examples

run checkdb
run checkdb -verbose
run checkdb -verbose -fix

run command script

Run an existing command script against a device or group of devices.

Synopsis

run command script [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-group <Groupname>] -name <Script Name> [-variables <Variable List>] [-linebyline] [-start <Task start date>] [-rep <Task repeat period>] [-sync] [-nowait] [-comment <Snapshop comment>] [-presnapshot <true or false>] [-postsnapshot <true, false or task>]

Description

Runs an existing command script, specified by name, against a device or group of devices. The proper variant of the script will be applied to each device. If no variant of the script supports a given device, that device will be skipped. The script is run as a system task.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-group - A valid group name. Either a device or a group must be specified, but not both (exactly one of -ip, -hostname, -fqdn or -group must be specified).
-name - Name of the command script to run
-variables - A list of variables to be replaced in the script - provided as a list of name=value pairs, separated by commas. Values can be surrounded in single-quotes ('). Within a quoted value, a single-quote can be embedded with two single-quote characters. Example: "variable1=value1,varable2='this is ''value 2'''"
-linebyline - Indicates that line by line deployment is preferred, rather than file-based deployment
-start - YYYY:MM:DD:HH:mm. The first date on which the task will run.
-rep - (#min | #:# | #days | #weeks | #months) where # is a positive integer. #:# is hours:minutes--the two integers do not have to be the same. Do not use this option with -sync.
-sync - Indicates the command should return only after the snapshot retrieval task is complete. Do not use this option with -rep or -start.
-nowait - Indicates that the task should not wait if there is another task currently running against the same device.
-comment - An optional comment about the snapshot.
-presnapshot - If false, this indicates that the snapshot that runs before the script should be skipped.
-postsnapshot - If false, this indicates that the snapshot that runs after the script should be skipped. If "task", this indicates that snapshot after the script should run as a separate task.

Return Type

String

Examples

run command script -ip 192.0.2.10 -name "Extended Ping" -variables "Target_IP=192.0.2.10" -start 2004:02:29:23:59 -rep 2days -comment "running extended ping"
run command script -group mygroup -name "Set Interface Description" -variables="interface=Ethernet1,description='provider ''MCI'',link id T207'" -linebyline -sync

run diagnostic

Run a diagnostic on a device.

Synopsis

run diagnostic [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-group <Group Name>] -diagnostic <Diagnostic Name> [-rep <Task repeat period>] [-start <Task start date>] [-sync] [-comment <Run script comment>] [-duration <Estimated duration of script task.>] [-sessionlog <true or false>] [-customname <Custom name>] [-customvalue <Custom value>] [-retryInterval <Retry count>] [-retryCount <Retry interval>]

Description

Run the specified diagnostic on a specified device either right away, or at some point in the future. The run diagnostic operation is actually a scheduled task.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-group - A name of a device group (mutually exclusive with -ip, -host, or -fqdn)
-diagnostic - A diagnostic to run. Built-in diagnostics are '@ProductAbbreviation@ Routing Table', '@ProductAbbreviation@ Interfaces' and '@ProductAbbreviation@ OSPF Neighbors'.
-rep - (#min | #:# | #days | #weeks | #months) where # is a positive integer. #:# is hours:minutes--the two integers do not have to be the same. Do not use this option with -sync.
-start - YYYY:MM:DD:HH:mm. The first date on which the task will run. Do not use this option with -sync.
-sync - Indicates that the command should return only after the deploy task is complete. Do not use this option with -start.
-comment - An optional comment about the diagnostic.
-duration - A number concatenated with a units signifier. Valid signifiers are m (minutes), h (hours), d (days), w (weeks). If this option is not provided, the duration for the task is set to 60 minutes.
-sessionlog - If true a complete session log will be saved with this task.
-customname - The custom field name.
-customvalue - The custom field value.
-retryInterval - The number of seconds between retries.
-retryCount - The number of times to retry the task if it fails.

Return Type

String

Examples

run diagnostic -ip 192.0.2.10 -diagnostic "vlan report" -sync
run diagnostic -ip 192.0.2.10 -diagnostic "@ProductAbbreviation@ Routing Table" -start 2004:02:29:23:59
run diagnostic -group "Core Routers" -diagnostic "@ProductAbbreviation@ OSPF Neighbors" -rep 7days -start 2004:01:01:01:00:00 -comment "Weekly Core Router OSPF Neighbors pull"

run external application

Execute a command.

Synopsis

run external application -app <Command> [-start <Task start date>] [-rep <Task repeat period>] [-sync] [-comment <Comment text>] [-startdir <Directory path>] [-resultfile <File path>] [-errorifnonzero <true or false>]

Description

Runs a @ProductAbbreviation@ task which spawns a new process that executes a command external to @ProductAbbreviation@.

-app - The command to execute.
-start - YYYY:MM:DD:HH:mm The time when the command will be executed. Do not use this option with -sync.
-rep - (#min | #:# | #days | #weeks | #months) where # is a positive integer. #:# is hours:minutes--the two integers do not have to be the same. Do not use this option with -sync.
-sync - Indicates that the CLI command should return only after the task is complete. Do not use this option with -start.
-comment - Comments to be attached to the task that runs to execute the command.
-startdir - The working directory of the process in which the command is executed.
-resultfile - The file to contain the output of the command.
-errorifnonzero - If true the task will be marked FAILED or WARNING if the command returns a non zero result code.

Return Type

String

Examples

run external application -start 2006:03:23:11:33 -startdir /usr/local/bin -resultfile /home/jdoe/out.log -app"echo foo"
run external application -app "grep -c /bin/csh /etc/passwd" -resultfile /home/jdoe/out.log -sync

run gc

Run the garbage collector.

Synopsis

run gc

Description

Recycle unused objects to increase the amount of free memory.

Return Type

String

Examples

run gc

run script

Run a command script on a device.

Synopsis

run script [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-group <Group Name>] [-mode <Command Script Mode>] -script <Command Script> [-rep <Task repeat period>] [-start <Task start date>] [-sync] [-nowait] [-comment <Run script comment>] [-presnapshot <true or false>] [-postsnapshot <true, false or task>] [-disablesessionlogging]

Description

Run the specified command script on a specified device either right away, or at some point in the future. The run script operation is actually a scheduled task. If no mode is specified the first supported enable, supervisor, provisioning or root mode will be used.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-group - A name of a device group (mutually exclusive with -ip, -host, or -fqdn)
-mode - A command script mode to run the script in.
-script - A script to run, may separate commands with '\n'. Commands that require multiple entries before returning to the device prompt can separate each entry with '\\r\\n'.

-rep - (#min | #:# | #days | #weeks | #months) where # is a positive integer. #:# is hours:minutes--the two integers do not have to be the same. Do not use this option with -sync.
-start - YYYY:MM:DD:HH:mm. The first date on which the task will run. Do not use this option with -sync.
-sync - Indicates that the command should return only after the deploy task is complete. Do not use this option with -start.
-nowait - Indicates that the task should not wait if there is another task currently running against the same device.
-comment - An optional comment about the script being run.
-presnapshot - If false, this indicates that the snapshot that runs before the script should be skipped.
-postsnapshot - If false, this indicates that the snapshot that runs after the script should be skipped. If "task", this indicates that snapshot after the script should run as a separate task.
-disablesessionlogging - Indicates that the session should not be logged.

Return Type

String

Examples

run script -ip 192.0.2.10 -mode "Cisco IOS enable" -script "show ver" -sync
run script -ip 192.0.2.10 -mode "Nortel BCC" -script "show system info" -start 2004:02:29:23:59
run script -group "Core Routers" -mode "Cisco IOS configuration" -script "banner motd xCore Router\\r\\ndo not touch!\nprompt %h%p" -start 2004:01:01:01:00:00 -comment "Get the core routers banner and prompt standardized"

show access

Display a device access record.

Synopsis

show access -id <Device access record ID>

Description

-id - Specifies a device access record. Think of this as a "device access record ID".

Return Type

DeviceAccessLogVO with columns:

accessTrigger
actionTaken
changeEventData
comments

configComments
configID
createDate
createUserID
deviceAccessLogID
deviceDataCustom1
deviceDataCustom2
deviceDataCustom3
deviceDataCustom4
deviceDataCustom5
deviceDataCustom6
deviceID
displayName
externalChangeRequestID
interceptorLogID
isConfigChange
lastModifiedDate
noPrune

Examples

show access -id 510

show acl

Show ACL.

Synopsis

show acl -id <Device ACL ID>

Description

Displays the device ACL in the system including Script and Application.

-id - List only ACLs with this deviceaclid

Return Type

DeviceACLVO with columns:

aCLID
aCLType
application
applicationState
comments
deviceACLID
deviceID
handle
lastModifiedAccessLogID

lastModifiedDate
lastModifiedUserID
mostRecent
script

Examples

show acl -id 241

show basicip

Show a BasicIP model.

Synopsis

show basicip [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-id <Config ID>]

Description

If the -ip flag is given, show the BasicIP model for the most recent config for the specified device.If the -id flag is given, show the BasicIP model for the specified config.Include either the -id or -ip option, but not both.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-id - A config ID

Return Type

String

Examples

show basicip -ip 192.0.2.10
show basicip -ip "East Site:192.0.2.10"
show basicip -id 73253

show cache info

Get information about the cache regions defined in @ProductAbbreviation@.

Synopsis

show cache info [-region <REGION_NAME>] [-id <USE_ID>] [-key <USE_KEY>] [-x <EXTENDED_STATS>]

Description

Display the active cache regions in @ProductAbbreviation@.

-region - Specify the name of the region in use

-id - Specify the id of the region

-key - Specify the key of the reqion

-x - Display extended cache stats for each region

Return Type

None

Examples

show cache info

show cache info -region REGION_NAME

show cache info -region REGION_NAME -x true

show config

Show the contents of a config.

Synopsis

show config -id <Config ID> [-mask]

Description

-id - The ID of a config
-mask - Mask out sensitive information such as device passwords

Return Type

String

Examples

show config -id 2600
show config -id 2405 -mask

show configlet

Show the configlet inbetween start and end pattern.

Synopsis

show configlet [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-start <Start Block>] [-end <End Block>] [-type <Helper>]

Description

-ip - List all device ports on the device with this IP address
-host - List all device ports on the device with this hostname
-fqdn - List all device ports on the device with this Fully Qualified Domain Name
-deviceid - List all device ports on the device with this device ID
-start - Block start pattern for the configlet
-end - Block end pattern for the congfiglet
-type - Type (helper) of the configlet

Return Type

String

Examples

show configlet -ip 1.2.3.4 -start webfarm -type C_POOL

show device

Show a device's properties.

Synopsis

show device [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-id <Device ID>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-id - A device ID

Return Type

Collection:DeviceVO with columns:

accessMethods
assetTag
changeEventData
comments

consoleIPAddress
consolePort
consoleRealmName
contact
createDate
deviceCustom1
deviceCustom2
deviceCustom3
deviceCustom4
deviceCustom5
deviceCustom6
deviceGroup1ID
deviceGroup2ID
deviceGroup3ID
deviceID
deviceName
deviceType
driverName
duplexMismatchDetected
excludeFromPoll
feedSource
firmwareVersion
flashMemory
freePorts
geographicalLocation
hierarchyLayer
hostName
lastAccessAttemptDate
lastAccessAttemptStatus
lastAccessSuccessDate
lastConfigChangeUserID
lastConfigPolicyCheckedDate
lastDuplexDataUpdate
lastImportDate
lastModifiedUserID
lastRecordModifiedDate
lastSnapshotAttemptDate
lastSnapshotAttemptStatus
lastSnapshotDate
lastSnapshotSuccessDate
lastTopologyDataUpdate
lastUsedAuthenticationID
latestSoftwareHistoryID
latestStartupRunningDiffer
managementStatus
memory
model
modemNumber
mostRecentConfigID
nATIPAddress
nATRealmName

performACLParsing
policyImportance
policyInCompliance
primaryFQDN
primaryIPAddress
processor
putInServiceDate
rOMVersion
serialNumber
siteID
softwareVersion
tFTPServerIPAddress
ticketNumber
totalPorts
unmanagedDate
uptime
uptimeStoredDate
vendor

Examples

show device -ip 192.0.2.10
show device -ip "East Site:192.0.2.10"
show device -id 527

show device config

Show the config most recently retrieved from the specified device.

Synopsis

show device config [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID

Return Type

String

Examples

show device config -ip 192.0.2.10
show device config -ip "East Site:192.0.2.10"

show device credentials

Show the current device credentials.

Synopsis

show device credentials [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

The passwords are populated in following algorithm: 1) If device has a LastAuthentication records, the passwords will be retrieved from that record, otherwise: 2) If device has device specific passwords defined(on Edit Device page), the device specific passwords will be returned 3) Otherwise, the first applicable device password rule will be used The return value may contain some or all of following attributes, depending on the device setting: usernamepasswordenable_passwordread_communitywrite_communitysnmpv3_usersnmpv3_authpasswordsnmpv3_privpassword

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID

Return Type

Map

Examples

show device credentials -ip 192.0.2.10
show device credentials -ip "East Site:192.0.2.10"

show device family

Show the family classification associated with the specified device.

Synopsis

show device family -ip <IP address>

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

Return Type

String

Examples

show device family -ip 192.0.2.10
show device family -ip "East Site:192.0.2.10"

show device latest diff

Show the difference between two configs most recently retrieved from the specified device.

Synopsis

show device latest diff [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID

Return Type

String

Examples

show device latest diff -ip 192.0.2.10
show device latest diff -ip "East Site:192.0.2.10"

show device template

Show a device template's properties.

Synopsis

show device template -templateid <Template Device ID>

Description

-templateid - A template device ID

Return Type

Collection:DeviceVO with columns:

accessMethods

assetTag
changeEventData
comments
consoleIPAddress
consolePort
consoleRealmName
contact
createDate
deviceCustom1
deviceCustom2
deviceCustom3
deviceCustom4
deviceCustom5
deviceCustom6
deviceGroup1ID
deviceGroup2ID
deviceGroup3ID
deviceID
deviceName
deviceType
driverName
duplexMismatchDetected
excludeFromPoll
feedSource
firmwareVersion
flashMemory
freePorts
geographicalLocation
hierarchyLayer
hostName
lastAccessAttemptDate
lastAccessAttemptStatus
lastAccessSuccessDate
lastConfigChangeUserID
lastConfigPolicyCheckedDate
lastDuplexDataUpdate
lastImportDate
lastModifiedUserID
lastRecordModifiedDate
lastSnapshotAttemptDate
lastSnapshotAttemptStatus
lastSnapshotDate
lastSnapshotSuccessDate
lastTopologyDataUpdate
lastUsedAuthenticationID
latestSoftwareHistoryID
latestStartupRunningDiffer
managementStatus

memory
model
modemNumber
mostRecentConfigID
nATIPAddress
nATRealmName
performACLParsing
policyImportance
policyInCompliance
primaryFQDN
primaryIPAddress
processor
putInServiceDate
rOMVersion
realmName
serialNumber
siteID
siteName
softwareVersion
tFTPServerIPAddress
ticketNumber
totalPorts
unmanagedDate
uptime
uptimeStoredDate
vendor

Examples

show device template -templateid 527

show device template config

Show the config most recently retrieved from the specified device template.

Synopsis

show device template config [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-templateid <Template ID>]

Description

-host - A valid hostname

-fqdn - A valid Fully Qualified Domain Name

-templateid - A device ID

Return Type

String

Examples

show device template config -templateid 801

show device template config -host test

show device template config variables

Show the variables defined in the config for specified device template.

Synopsis

show device template config variables -templateid <Template ID>

Description

-templateid - A device template ID

Return Type

Collection:CustomVariable

Examples

show device template config variables -templateid 801

show deviceinfo

Show a DeviceInformation model.

Synopsis

show deviceinfo [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-id <Config ID>]

Description

If the -ip flag is given, show the DeviceInformation model for the most recent config for the specified device.If the -id flag is given, show the DeviceInformation model for the specified config.Include either the -id or -ip option, but not both.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-id - A config ID

Return Type

String

Examples

show deviceinfo -ip 192.0.2.10
show deviceinfo -ip "East Site:192.0.2.10"
show deviceinfo -id 73253

show diagnostic

Show a diagnostic's results.

Synopsis

show diagnostic -id <Diagnostic ID>

Description

-id - A diagnostic ID

Return Type

String

Examples

show diagnostic -id 73253

show driver

Show the driver assigned to a device.

Synopsis

show driver [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID

Return Type

String

Examples

show driver -ip 192.0.2.10

show driver -host rtr5.vfm.lab

show event

Display the details of an event.

Synopsis

show event -id <event ID>

Description

-id - A valid event id

Return Type

EventVO with columns:

configPolicyID
eventClass
eventData
eventDate
eventDeviceID
eventID
eventTaskID
eventText
eventType
eventUserID
importance
lastModifiedDate
siteID
ticketNumber

Examples

show event -id 27

show group

Show all information for a group.

Synopsis

show group [-name <Group name>] [-id <Group id>]

Description

-name - The group name for whom information will be displayed
-id - The group id for whom information will be displayed

Return Type

DeviceGroupVO with columns:

comments
core
createDate
deviceCount
deviceGroup1ID
deviceGroup2ID
deviceGroup3ID
deviceGroupCustom1
deviceGroupCustom2
deviceGroupCustom3
deviceGroupCustom4
deviceGroupCustom5
deviceGroupCustom6
deviceGroupID
deviceGroupName
deviceViewID
devices
dynamicFilter
isDynamic
isParent
isPolicyScope
lastModifiedDate
managerUserID
managingCoreID
parentDeviceGroupID
realmName
shared

Examples

show group -name johnd
show group -id 5

show icmp

Show an ICMPTest model.

Synopsis

show icmp [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-id <Config ID>]

Description

If the -ip flag is given, show the ICMPTest model for the most recent config for the specified device.If the -id flag is given, show the ICMPTest model for the specified config.Include exactly one of the -id or -ip option.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-id - A config ID

Return Type

String

Examples

show icmp -ip 192.0.2.10
show icmp -id 73253

show int

Show a ShowInterfaces model.

Synopsis

show int [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-id <Config ID>]

Description

If the -ip flag is given, show the ShowInterfaces model for the most recent config for the specified device.If the -id flag is given, show the ShowInterfaces model for the specified config.Include either the -id or -ip option, but not both.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-id - A config ID

Return Type

String

Examples

show int -ip 192.0.2.10
show int -ip "East Site:192.0.2.10"
show int -id 73253

show ip

Show a ip's properties.

Synopsis

show ip -ipvalue <Value> [-deviceip <Device IP address>] [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-ipvalue - The ip value a.b.c.d where 0 <= a,b,c,d <= 255
-deviceip - The device's ip address a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID

Return Type

IpVO with columns:

changeDate
comments
deviceID
devicePortID
ipCustom1
ipCustom2
ipCustom3
ipCustom4
ipCustom5
ipCustom6
ipID
ipMask
ipName
ipPriority
ipType
ipValue
lastModifiedUserID
networkAddressStart
paddedIp

password
realmName
redirectLocation
subnetBitCount
usedToAccess
userName
variables

Examples

show ip -deviceip 192.0.2.10 -ipvalue 192.0.2.10
show ip -deviceid 1401 -ipvalue 192.0.2.10

show latest access

Show the most recent access record for the specified device.

Synopsis

show latest access [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID

Return Type

DeviceAccessLogVO with columns:

accessTrigger
actionTaken
changeEventData
comments
configComments
configID
createDate
createUserID
deviceAccessLogID
deviceDataCustom1
deviceDataCustom2
deviceDataCustom3
deviceDataCustom4
deviceDataCustom5
deviceDataCustom6
deviceID

displayName
externalChangeRequestID
interceptorLogID
isConfigChange
lastModifiedDate
noPrune

Examples

show latest access -ip 192.0.2.10
show latest access -ip "East Office:192.0.2.10"

show metadata

Show a specific piece of custom data.

Synopsis

show metadata -metadataid <Metadata ID>

Description

-metadataid - ID of the custom data to show

Return Type

MetadataVO with columns:

associatedTableID
data
fieldID
lastModifiedDate
metadataID

Examples

show metadata -metadataid 54535

show metadata field

Show a custom data field.

Synopsis

show metadata field -fieldid <Field ID>

Description

-fieldid - ID of the custom data field to show

Return Type

MetadataFieldVO with columns:

associatedTable
fieldDataType
fieldID
fieldName
fieldValues
flags
inUse
lastModifiedDate

Examples

show metadata field -fieldid 8394

show module

Show a module's properties.

Synopsis

show module -id <Module ID>

Description

-id - The ID of a module

Return Type

DeviceModuleVO with columns:

comments
deviceID
deviceModuleID
firmwareVersion
hardwareRevision
lastModifiedDate
memory
moduleCustom1
moduleCustom2
moduleCustom3
moduleCustom4
moduleCustom5
moduleCustom6
moduleDescription
moduleModel
moduleOS
serialNumber
slot

slotNumber

Examples

show module -id 527

show ospfneighbor

Show a ShowOSPFNeighbors model.

Synopsis

show ospfneighbor [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-id <Config ID>]

Description

If the -ip flag is provided, show the ShowOSPFNeighbors model for the most recent config for the specified device.If the -id flag is given, show the ShowOSPFNeighbors model for the specified config.Include either the -id or -ip option, but not both.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-id - A config ID

Return Type

String

Examples

show ospfneighbor -ip 192.0.2.10
show ospfneighbor -ip "East Site:192.0.2.10"
show ospfneighbor -id 73253

show permission

Display whether or not a user has permissions for a particular resource.

Synopsis

show permission -resource <resource> [-u <username>] [-id <user ID>]

Description

-resource - The name of a Command or Resource.

-u - Username
-id - User ID

Return Type

String

Examples

show permission -resource com.rendition.dib.DeleteACLTask -u bob
show permission -resource "Add Device" -id 101715

show policy

Shows policy information

Synopsis

show policy -id <Policy ID>

Description

-id - policy id

Return Type

ConfigPolicyVO with columns:

CVE
comments
configPolicyID
configPolicyName
createDate
description
disclosureDate
inUse
lastModifiedDate
lastModifiedUserID
scope
siteID
solution
status
tag
ticketNumber
vendorAdvisoryURL
vendorSolutionURL

Examples

show policy -id 6120

show policy compliance

Shows policies and device compliance states

Synopsis

show policy compliance [-policyid <Policy ID>] [-deviceid <Device ID>] [-compliance <Compliance State (in|out|unknown)>]

Description

-policyid - policy id
-deviceid - device id
-compliance - compliance state (in|out|unknown)

Return Type

Collection:ConfigPolicyComplianceVO with columns:

configPolicyID
configPolicyName
deviceID
inCompliance
lastModifiedDate

Examples

show policy compliance
show policy compliance -policyid 6120
show policy compliance -deviceid 312
show policy compliance -policyid 6120 -deviceid 312
show policy compliance -policyid 6120 -compliance in
show policy compliance -deviceid 25549 -compliance out

show policy rule

Shows rule information

Synopsis

show policy rule -id <Rule ID>

Description

-id - rule id

Return Type

ConfigRuleVO with columns:

appliesToEntireDeviceFamily
blockEndPattern
blockStartPattern
comments
conditions
configPolicyID
configRuleID
configRuleName
createDate
description
deviceFamily
evaluationLogic
importance
inUse
lastModifiedDate
lastModifiedUserID
ruleType
scope
ticketNumber

Examples

show policy rule -id 3508

show polling status

Show the current status of polling.

Synopsis

show polling status

Description

Return Type

String

Examples

show polling status

show port

Show a port's properties.

Synopsis

show port -id <Port ID>

Description

-id - The ID of a port

Return Type

DevicePortVO with columns:

associatedVlanID
comments
configuredDuplex
configuredSpeed
description
deviceID
devicePortID
ipAddresses
lastModifiedDate
macAddress
negotiatedDuplex
negotiatedSpeed
portAllows
portCustom1
portCustom2
portCustom3
portCustom4
portCustom5
portCustom6
portName
portState
portStatus
portType
slotNumber
temporaryVlanName

Examples

show port -id 527

show routing

Display a routing table.

Synopsis

show routing [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-id <Routing table ID>]

Description

If the -ip flag is given, show the most recent routing table captured for the specified device.If the -id flag is given, show the specified routing table.Include either the -id or -ip option, but not both.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID
-id - A routing table ID

Return Type

String

Examples

show routing -host rtr6.vfm.lab
show routing -id 3276

show rule condition

Shows rule condition information

Synopsis

show rule condition -id <Condition ID>

Description

-id - condition id

Return Type

RuleConditionVO with columns:

comments
configRuleID
createDate
exactOrder
lastModifiedDate
modelDataName
operand
operatorName
ruleConditionID
ruleConditionOrder
useRegexp

Examples

show rule condition -id 3508

show script

Show one command script, advanced script or diagnostic.

Synopsis

show script [-id <Script / Diagnostic ID>] [-name <Script / Diagnostic Name>] [-type <Script / Diagnostic Type>]

Description

Output the indicated command script, advanced script or diagnostic. The desired script or diagnostic can be specified by ID, or by a combination of name and type. If more than one name match occurs, then an error will be reported and you must specify the unique script desired by ID.

-id - ID of the desired script or diagnostic
-name - Name of the desired script or diagnostic
-type - Type of the desired script or diagnostic - may be command, advanced or diagnostic

Return Type

CustomScriptVO with columns:

createDate
createUserID
customScriptID
description
lastModifyDate
lastModifyUserID
name
parameters
script
scriptMode
scriptType
siteID
taskType
variableData

Examples

show script -id 5
show script -name "Edit Port Duplex" -type command

show server option

Display the setting of a server option.

Synopsis

show server option -name <option name> [-default <default value>]

Description

Display the value of an Admin Setting or server configuration option. If the option is not set and no default is provided then this command will fail.

-name - The name of the server option.
-default - The value to return if the option is not set.

Return Type

String

Examples

show server option -name proxy/ssh_listener_port
show server option -name customer/https_port -default @HTTPS_PORT@

show service type

Shows service types assigned to a device

Synopsis

show service type [-ip <IP Address>] [-deviceid <Device ID>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-deviceid - device id

Return Type

Collection:DevicePropertyVO with columns:

mCreateDate
mCustomServiceTypes
mDeviceID
mDevicePropertyID
mLastModifiedDate
mServiceTypes

Examples

show service type -deviceid 3508

show session

Show interceptor log record.

Synopsis

show session -id <Interceptor log id>

Description

-id - Interceptor log ID

Return Type

InterceptorLogVO with columns:

deviceID
deviceIP
endDate
interceptorLogCustom1
interceptorLogCustom2
interceptorLogCustom3
interceptorLogCustom4
interceptorLogCustom5
interceptorLogCustom6
interceptorLogID
lastModifiedDate
m_userName
sessionData
sessionType
startDate
status
userID

Examples

show session -id 5

show session commands

List all commands in interceptor log record.

Synopsis

show session commands -id <Interceptor log id>

Description

-id - Interceptor log ID

Return Type

String

Examples

show session commands -id 5

show snapshot

Show the config most recently retrieved from the specified device.

Synopsis

show snapshot [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name
-deviceid - A device ID

Return Type

String

Examples

show snapshot -ip 192.0.2.10
show snapshot -ip "East Site:192.0.2.10"

show system message

Display the details of a system message.

Synopsis

show system message -id <System message ID>

Description

-id - A valid system message id

Return Type

EventVO with columns:

configPolicyID
eventClass
eventData
eventDate
eventDeviceID
eventID
eventTaskID
eventText
eventType
eventUserID
importance
lastModifiedDate
siteID
ticketNumber

Examples

show system message -id 27

show task

Shows detailed information about a task.

Synopsis

show task -id <Task ID>

Description

-id - The task ID whose details will be displayed

Return Type

ScheduleTaskVO with columns:

approvalPriority
approvalStatus
approvalUsers
approveByDate
comments
completeDate
coreID
createDate
csvData
csvGroupTask
deviceDataID
deviceGroupID

deviceID
duration
expensive
failedChildCount
failureType
lastModifiedDate
parentTaskID
pendingChildCount
repeatCount
repeatEndDate
repeatInterval
repeatType
repeatWeekday
reservationEndDate
reservedDuration
reservedDurationType
result
resultConfigID
retryCount
retryInterval
runASAP
scheduleDate
scheduleTaskID
siteID
startDate
status
subTask
subTasks
succeededChildCount
systemTask
taskCustom1
taskCustom2
taskCustom3
taskCustom4
taskCustom5
taskCustom6
taskData
taskName
taskType
taskUserID
ticketNumber
userGeneratedChange

Examples

show task -id 354

show topology

Show details for a single topology record.

Synopsis

show topology -id <Topology Data ID>

Description

-id - The id of the topology record to show.

Return Type

DeviceTopologyDataVO with columns:

crossReferenceTopologyDataID
data
deviceID
devicePortID
deviceTopologyDataID
firstSeen
lastSeen
lastUpdated
remoteDeviceID
remoteDevicePortID
remoteSasServerID
remoteSasServerInterfaceID
topologyDataType

Examples

show topology -id 6543201

show user

Show all information for a user.

Synopsis

show user [-u <User name>] [-id <User id>]

Description

-u - The user name for whom information will be displayed \\n NOTE: If user name contains '\' escape charater, the input needs to be escaped properly. For example, if user name is similar to "Domain\User", the input needs to be "Domain\\User"
-id - The user id for whom information will be displayed

Return Type

UserVO with columns:

aaaPassword
aaaUserName
allowFailover
comments
createDate
deviceGroup1ID
deviceGroup2ID
deviceGroup3ID
distinguishedName
emailAddress
firstName
lastLoginDate
lastModifiedDate
lastName
privilegeLevel
requiredUser
status
ticketNumber
timeZone
useAaaLoginForProxy
userCustom1
userCustom2
userCustom3
userCustom4
userCustom5
userCustom6
userID
userName
userPassword

Examples

show user -u johnd
show user -id 5

show version

Display the release version of @ProductAbbreviation@.

Synopsis

show version

Description

Display the release version of @ProductAbbreviation@.

Return Type

String

Examples

show version

show vlan

Shows Vlan information for a given Vlan ID for a device

Synopsis

show vlan [-deviceip <Device IP address>] [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] -vlanid <Vlan ID>

Description

-deviceip - The device's ip address a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-host - A valid hostname

-fqdn - A valid Fully Qualified Domain Name

-deviceid - A device ID

-vlanid - Vlan ID

Return Type

DeviceVlanInfoVO with columns:

description
deviceID
deviceVlanInfoID
lastModifiedDate
lastModifiedUserID
mtu
vlanID
vlanName
vlanPrivate
vlanStatus
vlanType

Examples

show vlan -deviceid 1301 -vlanid 221
show vlan -ip 1.2.3.4 -vlanid 221

show vtp

Shows VTP information for a specific device

Synopsis

show vtp [-deviceip <Device IP address>] [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>]

Description

-deviceip - The device's ip address a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.

-host - A valid hostname

-fqdn - A valid Fully Qualified Domain Name

-deviceid - A device ID

Return Type

DeviceVTPVO with columns:

configurationVersion
currentVLANNumber
description
deviceID
deviceVTPID
domainName
lastModifiedDate
lastModifiedUserID
maxVLANNumber
md5Digest
operatingMode
pruningMode
vtpTrapsGeneration
vtpV2Mode
vtpVersion

Examples

show vtp -deviceid 201

source

Have the the system client execute all commands contained within a text file.

Synopsis

source <The name of the file containing CLI commands to execute.>

Description

This command has no options but takes one argument: the name of the file to "source". The source file should contain only valid CLI commands each seperated by one newline.

Return Type

None

Examples

```
source C:\temp\commands.txt
```

ssh

Make an ssh connection to a device.

Synopsis

ssh [-override] []

Description

Connect to a device through the system's Proxy Interface via ssh (bypassing single sign-on). If you are connected to a device through a console server, you may hit ctrl-\ to return to the system shell after logging out of the device.

-override - Force a connection to a device in the event that simultaneous connection warning or prevention is turned on.
- Hostname, Device ID, Fully Qualified Domain Name, or Primary IP Address to use to lookup the device to connect to. The characters * and ? can be used as wildcards.
- Port to use to connect to devices outside of the system.

Return Type

None

Examples

```
ssh 192.0.2.10
ssh -override mydevice
```

stop task

Stop a running task.

Synopsis

stop task -id <Task ID>

Description

-id - The task ID of the task to stop.

Return Type

String

Examples

stop task -id 54

stop task all

Stop all Running and Waiting tasks.

Synopsis

stop task all

Description

Return Type

String

Examples

stop task all

synchronize

Synchronize a device's startup and running configs.

Synopsis

synchronize [-ip <IP address>] [-host <Hostname>] [-fqdn <Fully Qualified Domain Name>] [-deviceid <Device ID>] [-group <Group Name>] [-skipinsync <Skip if Synchronized>] [-rep <Task repeat period>] [-start <Task start date>] [-sync] [-comment <Task comment>]

Description

Synchronize a device's startup configuration so it matches its running configuration. The synchronize operation is actually a scheduled task.

-ip - a.b.c.d where 0 <= a,b,c,d <= 255. You may optionally prefix the IP with SITE: where SITE is the name of the Site the device is in.
-host - A valid hostname
-fqdn - A valid Fully Qualified Domain Name

-deviceid - A device ID
-group - A name of a device group (mutually exclusive with -ip, -host, or -fqdn)
-skipinsync - Indicates that the command should skip any device that the system indicates already has matching startup and running configs.
-rep - (#min | #:# | #days | #weeks | #months) where # is a positive integer. #:# is hours:minutes--the two integers do not have to be the same. Do not use this option with -sync.
-start - YYYY:MM:DD:HH:mm. The first date on which the task will run. Do not use this option with -sync.
-sync - Indicates that the command should return only after the synchronize task is complete. Do not use this option with -start.
-comment - An optional comment about the synchronize task.

Return Type

String

Examples

synchronize -ip 192.0.2.10 -sync
synchronize -ip 192.0.2.10 -start 2004:02:29:23:59
synchronize -group "Core Routers" -skipinsync -start 2004:01:01:01:00:00 -comment "Make sure core routers have matching startup and running configs"

telnet

Make a telnet connection to a device.

Synopsis

telnet [-override] []

Description

Connect to a device through the system's Proxy Interface via telnet (bypassing single sign-on). If you are connected to a device through a console server, you may hit ctrl-\ to return to the system shell after logging out of the device.

-override - Force a connection to a device in the event that simultaneous connection warning or prevention is turned on.
- Hostname, Device ID, Fully Qualified Domain Name, or Primary IP Address to use to lookup the device to connect to. The characters * and ? can be used as wildcards.
- Port to use to connect to devices outside of the system.

Return Type

None

Examples

telnet 192.0.2.10
telnet -override mydevice

test config

Test policy compliance for a device configuration script.

Synopsis

test config -family <Device Family> -script <Configuration Script> [-policy <Policy Name>] [-group <Device Group>]

Description

This command is used to verify whether a configuration script is in compliance with applicable policies.

-family - The device family for the configuration script to be tested("Cisco IOS", F5, etc.)
-script - The configuration script to be tested.
-policy - The name of the policy for which the script will be test against.
-group - Specify a device group name. The test will be performed against the policies that are applicable to the group. If both -policy and -group are used, -group argument will be ignored.If none of -policy and -group is used, test will be performed against all applicable policies.

Return Type

String

Examples

test config -family "Cisco IOS" -script "version 12.1 ...."
* Note this command is intended for API use since it is difficult to input the entire configuration script in the command line.

test software

Test software levels for a device or device group.

Synopsis

test software [-ip <IP Address>] [-group <Device Group>]

Description

-ip - The IP address of a single device to test.

-group - A device group containing multiple devices to test.

Return Type

String

Examples

test software -ip 192.0.2.10
test software -group CoreRouters

traceroute

Run a traceroute command on a device.

Synopsis

traceroute -source <IP address | Hostname | Fully Qualified Domain Name> -sourcegroup <Group name> -dest <List of devices> -rep <Task repeat period> -async -start <task start date>

Description

Causes a series of traceroute commands to be executed on a device. One traceroute command is executed for each target host specified. This series of commands may by run on the device immediately, or scheduled to run sometime in the future. Via this command, the task scheduled can be set to repeat periodically. Note that if not scheduled as a task, this command may take some time to complete.

-source - Can be an IP address (a.b.c.d where 0 <= a,b,c,d <= 255), or a valid hostname, a valid Fully Qualified Domain Name.
-sourcegroup - A valid group name. Exactly one of -source or -sourcegroup must be specified.
-dest - A comma seperated list of devices. Devices may be specified in any way that is understood by the traceroute program on the device specified by the option "-source".
-rep - (#min | #:# | #days | #weeks | #months) where # is a positive integer. #:# is hours:minutes, the two integers don't have to be the same. This option should not be used unless -async is also supplied.
-async - Indicates that the traceroute operation should be scheduled on the system as a task. The start time for the task will be immediatly unless an alternate start data is provided by means of the -start option.
-start - YYYY:MM:DD:HH:mm. The date on which the task will first be run. This option should not be used unless -async is also supplied.

Return Type

String

Examples

traceroute -source 192.0.2.10 -dest 192.0.2.10
traceroute -source 192.0.2.10 -dest 192.0.2.10,192.0.2.10,192.0.2.10
traceroute -source 192.0.2.10 -dest 192.0.2.10 -start 2004:02:29:23:59 -rep 2days
traceroute -source 192.0.2.10 -dest 192.0.2.10 -async
traceroute -sourcegroup mygroup -dest 192.0.2.10

undeploy image

Undeploy software images from device.

Synopsis

undeploy image -ip <device ip address> -images <images separated by ,> [-reboot <reboot instruction>] [-rebootwait <reboot wait (in seconds)>] [-filesystem <file system of device>] [-pretask <task to run before delete>] [-posttask <task to run after delete>] [-start <Task start date>] [-comment <Snapshot comment>] [-duration <Estimated duration of snapshot task.>] [-sessionlog <true or false>] [-customname <Custom name>] [-customvalue <Custom value>] [-retryInterval <Retry count>] [-retryCount <Retry interval>]

Description

Delete software images from device.

-ip - ip address of the device the images will be deleted.
-images - images to be deleted.
-reboot - wheather to reboot the device after deleting images.
-rebootwait - seconds to wait before reboot.
-filesystem - name of filesystem of the device the images will be deleted.
-pretask - name of task before delete.
-posttask - name of task after delete.
-start - YYYY:MM:DD:HH:mm. The first date on which the task will run. The string "now" means the current time. The string "tomorrow" means 24 hours from the current time.
-comment - An optional comment about the snapshot.
-duration - A number concatenated with a units signifier. Valid signifiers are m (minutes), h (hours), d (days), w (weeks). If this option is not provided, the duration for the task is set to 60 minutes.
-sessionlog - If true a complete session log will be saved with this task.
-customname - The custom field name.
-customvalue - The custom field value.
-retryInterval - The number of seconds between retries.
-retryCount - The number of times to retry the task if it fails.

Return Type

String

Examples

undeploy image -ip 10.1.1.1 -images bar.bin,baz.bin -filesystem flash:
undeploy image -ip 10.1.1.1 -images bar.bin,baz.bin -filesystem flash: -reboot -rebootwait 60
undeploy image -ip 10.1.1.1 -images bar.bin,baz.bin -filesystem flash: -reboot -rebootwait 60 -posttask squeeze

update dynamic group

Update dynamic group's member devices.

Synopsis

update dynamic group -name <Group name>

Description

Recalculate a dynamic group's member devices based on the predefined criteria. This has no effect on a non-dynamic device group.

-name - The group name for which the member devices will be updated

Return Type

Status

Examples

update dynamic group -name "all device out of compliance"

version

Display the system version.

Synopsis

version

Description

Displays the system version build number.

Return Type

None

Examples

version

Appendix B: Installing the Perl API

There are two methods for installing the Perl API:

Auto Installer method

Manual Install method

Auto Installer Method

The Auto Installer installs all of the Perl API modules, as well as their dependencies.

Open a shell. If you are on a Windows platform, open a command prompt. If you are on a Linux or Solaris platform, you can either open a command shell or SSH into the NCM server.

Note: You will need privileges to both create and modify files for NCM and Perl. As a result, you might need Administrator privileges on a Windows Platform and root privileges on Linux or Solaris platforms.

Change to the directory where NCM is installed. This directory will have been set when you installed NCM.

To run the install script, enter: `perl client/perl_api/har/install.pl`

Note: If Perl is not in your path and/or you have multiple Perl versions installed, use the full path to the Perl executable that you will be using. This should also match the value for the Perl interpreter set in the NCM server configuration.

This procedure installs all of the Perl API modules, as well as their dependencies. However, only "pure perl" dependencies are provided. For example, SOAP::Lite is provided, which includes a minimalist lightweight XML parser. For the best performance, it is recommended that you have the XML::Parser module installed.

If you are using ActivePerl (with a Perl version of 5.8 or better), the XML::Parser module is included with the distribution. Otherwise, you will need to use PPM, CPAN, or manually download and install the module.

Manual Install Method

Confirm that certain versions of Perl and/or Perl modules (that are not part of some core Perl distributions) are installed before you begin. See the META.yml file within each package/tarball for its requirements.

If your Perl distribution does not contain all of the required Perl modules, they are available at http://www.cpan.org and/or via PPM. (If you are using ActivePerl, try PPM first.)

To install any of the required modules, use one of the following commands:

ppm install SOAP-Lite

cpan install SOAP::Lite

Note that PPM (ppm.exe) is part of the ActivePerl distribution. If you are using ActivePerl, it is recommended that you use the PPM method. You can also run PPM without arguments and then issue the install command. You may need to do this for some Perl modules that have multiple versions to choose from, followed by install # where # is the item in the list returned by the install command. Keep in mind that PPM prefers to use the '-' as a namespace separator in place of the Perl '::' separator.

Note: NMAKE.EXE is installed when installing NCM on a Windows platform. It is located the /client directory. CPAN is simply a wrapper for the perl -MCPAN -e shell command. The CPAN command (or cpan.exe) is part of the core Perl install on all Perl versions since 5.8.0 (including ActivePerl).

Keep in mind that the installation could fail if your Perl installation does not meet certain requirements. See the “Installation Requirements” section. In addition, the Perl modules are distributed as compressed tarballs, similar modules on CPAN. They are located in <NCM_ROOT>/client/perl_api/.

To untar and uncompress all of the modules at one time, use the ptar command. ptar is distributed as part of the popular Perl module Archive::Tar, which is included in the standard ActivePerl distributions. To view the contents of the directory and to extract the contents into your current directory, enter: ptar -xzvf PATH/TO/whatever.tar.gz.

To install the Perl API on a Windows platform with ActivePerl (or any platform running a version of Perl that has the Module::Build module installed):

```
perl Build.PL

perl Build build

perl Build test

perl Build install
```

You may also use the traditional CPAN method. Enter:

```
perl Makefile.PL

make

make test

make install
```

Note: If you are using the CPAN method on a Windows platform, you will need to enter nmake rather than make.