Cisco Connected Mobile Experiences Command Reference Guide

Release 10.1
May, 2015

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

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

Cisco Connected Mobile Experiences Command Reference Guide

CONTENTS

Preface v

CHAPTER iii CLI Commands iii-1

cmxctl Commands iii-1

cmxctl checklogs iii-2

cmxctl debug iii-3

cmxctl disable iii-4

cmxctl enable iii-5

cmxctl jobs iii-6

cmxctl node iii-7

cmxctl restart iii-8

cmxctl start iii-9

cmxctl status iii-11

cmxctl stop iii-13

cmxctl version iii-14

cmxctl config Commands iii-15

cmxctl config backup iii-16

cmxctl config controllers iii-17

cmxctl config import iii-18

cmxctl config maps iii-19

cmxctl config reload iii-20

cmxctl config restore iii-21

cmxctl config verify iii-22

Preface

This preface describes the audience for, organization of, and conventions used in this document. It also provides information about how to obtain related documentation. This chapter includes the following sections:

- Audience, page v
- Related Documentation, page vi
- Obtaining Documentation and Submitting a Service Request, page vii

Audience

This document is for experienced network administrators who will configure and maintain Connected Mobile Experiences (Cisco CMX).

Conventions

This document uses the following conventions:

Convention	Indication
bold font	Commands, keywords, and user-entered text appear in **bold** font.
italic font	Document titles, new or emphasized terms, and arguments for which you supply values are in *italic* font.
[ ]	Elements in square brackets are optional.
{x I y I z}	Required alternative keywords are grouped in braces and separated by vertical bars.
[x I y I z]	Optional alternative keywords are grouped in brackets and separated by vertical bars.
string	A nonquoted set of characters. Do not use quotation marks around the string. If you do the string will include the quotation marks.
`courier` font	Terminal sessions and information the system displays appear in `courier` font.
< >	Nonprinting characters such as passwords are in angle brackets.

[ ]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

Note Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the manual.

Tip Means *the following information will help you solve a problem*. The tips information might not be troubleshooting or even an action, but could be useful information, similar to a Timesaver.

Caution Means *reader be careful*. In this situation, you might perform an action that could result in equipment damage or loss of data.

Timesaver Means *the described action saves time*. You can save time by performing the action described in the paragraph.

Warning **IMPORTANT SAFETY INSTRUCTIONS**

This warning symbol means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Use the statement number provided at the end of each warning to locate its translation in the translated safety warnings that accompanied this device.

SAVE THESE INSTRUCTIONS

Warning **Statements using this symbol are provided for additional information and to comply with regulatory and customer requirements.**

Related Documentation

For more information about Cisco Mobility Services Engine and related products, see: http://www.cisco.com/c/en/us/support/wireless/mobility-services-engine/tsd-products-support-series-home.html

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation* which also lists all new and revised Cisco technical documentation, at:
http://www.cisco.com/c/en/us/td/docs/general/whatsnew/whatsnew.html.

Subscribe to *What's New in Cisco Product Documentation*, as an RSS feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service. Cisco currently supports RSS Version 2.0.

CHAPTER 1

CLI Commands

This chapter describes the Cisco Connected Mobile Experiences (Cisco CMX) CLI commands:

- cmxctl Commands, page 1-1
- cmxctl config Commands, page 1-15

Note In the context of Cisco Connected Mobile Experiences (Cisco CMX), to access the CLI you use either console to the VM or appliance or SSH to the server when it has IP connectivity.

cmxctl Commands

This section lists the **cmxctl** commands that you can use to perform debug, backup, and restore functionalities:

- cmxctl checklogs, page 1-2
- cmxctl debug, page 1-3
- cmxctl disable, page 1-4
- cmxctl enable, page 1-5
- cmxctl jobs, page 1-6
- cmxctl node, page 1-7
- cmxctl restart, page 1-8
- cmxctl start, page 1-9
- cmxctl status, page 1-11
- cmxctl stop, page 1-13
- cmxctl version, page 1-14

cmxctl checklogs

To check logs and generate a report, use the **cmxctl checklogs** command.

cmxctl checklogs

Syntax Description

This command has no arguments or keywords.

Command Default

None

Usage Guidelines

After a report is generated, the specific log that shows the error can be viewed for additional details.

For example, /opt/cmx.var.log.cmxjobs.log.3 has 108 errors. You can use the command more /opt/cmx.var.log.cmxjobs.log.3 to view that file.

Examples

The following example shows how to check logs and generate a report:

```
[root@server]# cmxctl checklogs

************************************************************************
Checking /opt/cmx/var/log/cmxjobs.log.3 for errors..
/opt/cmx/var/log/cmxjobs.log.3 has 108 errors
**********************************************************************
Checking /opt/cmx/var/log/system-cron.log for errors..
/opt/cmx/var/log/system-cron.log has 0 errors
************************************************************************
Checking /opt/cmx/var/log/cmxjobs.log for errors..
/opt/cmx/var/log/cmxjobs.log has 81 errors
************************************************************************
Checking /opt/cmx/var/log/collectd.log for errors..
/opt/cmx/var/log/collectd.log has 0 errors
***********************************************************************
Checking /opt/cmx/var/log/consul.log for errors..
/opt/cmx/var/log/consul.log has 0 errors
************************************************************************
Checking /opt/cmx/var/log/qless-py-worker.log for errors..
/opt/cmx/var/log/qless-py-worker.log has 0 errors
************************************************************************
Checking /opt/cmx/var/log/influxdb.log for errors..
/opt/cmx/var/log/influxdb.log has 0 errors
************************************************************************
Checking /opt/cmx/var/log/cmxjobs.log.4 for errors..
/opt/cmx/var/log/cmxjobs.log.4 has 108 errors
***********************************************************************
```

cmxctl debug

To create a debug tarball in the current directory, use the **cmxctl debug** command.

cmxctl debug

Syntax Description

This command has no arguments or keywords.

Command Default

None

Usage Guidelines

The debug tarball that is created will be approximately 300 MB and takes at 90 seconds to complete.

Examples

The following example shows how to create a debug tarball in the current directory:

```
[root@server]# cmxctl debug

running locally
Dumping debug information...
[localhost] Executing task 'dump_config'
cp: cannot stat `/opt/cmx/share/upgrade.answers': No such file or directory
[localhost] Executing task 'dump_state'
running 'ps aux'
running 'ifconfig -a'
running 'cmxctl status'
running 'ulimit -a'
running 'ps -u root,postgres -o %cpu,%mem,cmd'
running 'netstat -o -n -a'
running 'df -h'
running 'ntpdate -d 172.19.28.250'
running 'consul members'
[localhost] Executing task 'dump_apis'
getting /api/config/v1/clusters
getting /api/config/v1/nodes
[localhost] Executing task 'dump_hosts'
pinging configuration.service.consul
pinging location.service.consul
pinging 6379.cache.service.consul
pinging 6380.cache.service.consul
pinging 6381.cache.service.consul
pinging database.service.consul
pinging analytics.service.consul
pinging halo.service.consul
Done.
```

cmxctl disable

To disable a service, use the **cmxctl disable** command.

cmxctl disable {**analytics** | **agent** | **cache_6379** | **cache_6380** | **cache_6381** | **cassandra** | **configuration** | **confd** | **consul** | **database** | **haproxy** | **location** | **matlabengine** | **metrics** | **nmsplb** | **influxdb** | **iodocs** | **qlesspyworker**}

Syntax Description

analytics	Performs analytics on calculated location data.
agent	Manages CMX system lifecycle. starts, stops, and monitors all the services running in CMX.
cache_6379	Caches the service used by location service.
cache_6380	Caches the service used by analytics service.
cache_6381	Caches the service used by analytics service.
cassandra	Enables cassandra database service used by the location service for historical data.
configuration	Configures nodes and clusters.
confd	Internal service.
consul	Internal service.
database	Enables the database service used by analytics and configuration services.
haproxy	Enables the TCP or HTTP load balancer gateway to all service APIs.
location	Provides location service to compute location.
matlabengine	Provides access point heatmap for location service.
metrics	Collects system metrics.
nmsplb	Enables load balancer service used for distributing Network Mobility Services Protocol (NMSP) messages to location services.
influxdb	Enables database services used for storing statistics from various services.
iodocs	Enables online document service for REST API offered by various services.
qlesspyworker	Internal service.

Command Default

None

Examples

The following example shows how to disable the cassandra database service:

```
[root@server]# cmxctl disable cassandra

Done
The nodeagent service is currently running with PID: 31776
Stopping cassandra process...
Done
Successfully shutdown cassandra Process.
```

cmxctl enable

To enable a service, use the **cmxctl enable** command.

cmxctl enable {**analytics** | **agent** | **cache_6379** | **cache_6380** | **cache_6381** | **cassandra** | **configuration** | **confd** | **consul** | **database** | **haproxy** | **location** | **matlabengine** | **metrics** | **nmsplb** | **influxdb** | **iodocs** | **qlesspyworker**}

Syntax Description

analytics	Performs analytics on calculated location data.
agent	Manages CMX system lifecycle. starts, stops, and monitors all the services running in CMX.
cache_6379	Caches the service used by location service.
cache_6380	Caches the service used by analytics service.
cache_6381	Caches the service used by analytics service.
cassandra	Enables cassandra database service used by the location service for historical data.
configuration	Configures nodes and clusters.
confd	Internal service.
consul	Internal service.
database	Enables the database service used by analytics and configuration services.
haproxy	Enables the TCP or HTTP load balancer gateway to all service APIs.
location	Provides location service to compute location.
matlabengine	Provides access point heatmap for location service.
metrics	Collects system metrics.
nmsplb	Enables load balancer service used for distributing Network Mobility Services Protocol (NMSP) messages to location services.
influxdb	Enables database services used for storing statistics from various services.
iodocs	Enables online document service for REST API offered by various services.
qlesspyworker	Internal service.

Command Default

None

Examples

The following example shows how to enable analytics service:

```
[root@server]# cmxctl enable analytics

The nodeagent service is not running.
Agent is not running, starting it now.
Starting nodeagent Process...
Retrying..
Done
Started nodeagent service with PID: 31027
```

cmxctl jobs

To configure recurring background jobs, use the **cmxctl jobs** command.

cmxctl jobs {**cancel** *jobname* | **list** | **run** *jobname* | **runnow** *jobname*}

Syntax Description

cancel	Cancels a scheduled job.
jobname	Name of the job.
list	Lists all the scheduled jobs.
run	Runs a job at a specified time.
runnow	Triggers a one-time run of the job.

Command Default

None

Examples

The following example shows how to run a background job:

```
[root@server]# cmxctl jobs run LocationIndexCleanup

submitted the job, verify using cmxctl jobs list.
```

cmxctl node

To manage node installation, use the **cmxctl node** command.

cmxctl node {addswap | configure | install | reinstall | sslmode {disable | enable} | uninstall | upgrade | verify}

Syntax Description

addswap	Adds a 10GB swap space to the node.
configure	Confirms node specifications.
install	Installs the Cisco Connected Mobile Experiences (Cisco CMX) for the first time.
reinstall	Reinstalls the existing installation.
sslmode disable	Disables Secure Sockets Layer (SSL).
sslmode enable	Enables SSL.
uninstall	Uninstalls the software.
upgrade	Upgrades the Cisco CMX from a URL or a file.
verify	Verifies node configuration.

Command Default

None

Examples

The following example shows how to manage node installation:

```
[root@server]# cmxctl node sslmode enable

enabling ssl
ssl enabled
```

cmxctl restart

To restart a service, use the **cmxctl restart** command.

cmxctl restart {**analytics** | **agent** | **cache_6379** | **cache_6380** | **cache_6381** | **cassandra** | **configuration** | **confd** | **consul** | **database** | **haproxy** | **location** | **matlabengine** | **metrics** | **nmsplb** | **influxdb** | **iodocs** | **qlesspyworker**}

Syntax Description

analytics	Performs analytics on calculated location data.
agent	Manages CMX system lifecycle. starts, stops, and monitors all the services running in CMX.
cache_6379	Caches the service used by location service.
cache_6380	Caches the service used by analytics service.
cache_6381	Caches the service used by analytics service.
cassandra	Enables cassandra database service used by the location service for historical data.
configuration	Configures nodes and clusters.
confd	Internal service.
consul	Internal service.
database	Enables the database service used by analytics and configuration services.
haproxy	Enables the TCP or HTTP load balancer gateway to all service APIs.
location	Provides location service to compute location.
matlabengine	Provides access point heatmap for location service.
metrics	Collects system metrics.
nmsplb	Enables load balancer service used for distributing Network Mobility Services Protocol (NMSP) messages to location services.
influxdb	Enables database services used for storing statistics from various services.
iodocs	Enables online document service for REST API offered by various services.
qlesspyworker	Internal service.

Command Default

None

Examples

The following example shows how to restart services:

```
[root@server bin]# cmxctl restart database

Done
The nodeagent service is currently running with PID: 16718
Stopping postgres Process...
Successfully shutdown postgres Process.
Starting postgres Process...
Done
Started postgres service with PID: 25702
Exception while notifying CE
```

cmxctl start

To start a Cisco Connected Mobile Experiences (Cisco CMX) services, use the **cmxctl start** command.

cmxctl start {**analytics** | **agent** | **cache_6379** | **cache_6380** | **cache_6381** | **cassandra** | **configuration** | **confd** | **consul** | **database** | **haproxy** | **location** | **matlabengine** | **metrics** | **nmsplb** | **influxdb** | **iodocs** | **qlesspyworker**}

Syntax Description

analytics	Performs analytics on calculated location data.
agent	Manages CMX system lifecycle. starts, stops, and monitors all the services running in CMX.
cache_6379	Caches the service used by location service.
cache_6380	Caches the service used by analytics service.
cache_6381	Caches the service used by analytics service.
cassandra	Enables cassandra database service used by the location service for historical data.
configuration	Configures nodes and clusters.
confd	Internal service.
consul	Internal service.
database	Enables the database service used by analytics and configuration services.
haproxy	Enables the TCP or HTTP load balancer gateway to all service APIs.
location	Provides location service to compute location.
matlabengine	Provides access point heatmap for location service.
metrics	Collects system metrics.
nmsplb	Enables load balancer service used for distributing Network Mobility Services Protocol (NMSP) messages to location services.
influxdb	Enables database services used for storing statistics from various services.
iodocs	Enables online document service for REST API offered by various services.
qlesspyworker	Internal service.

Command Default

None

Examples

The following example shows how to start CMX services:

```
[root@server]# cmxctl start consul

Done
The nodeagent service is currently running with PID: 16718
Done
The analytics service is already running with pid: 1099
Done
Exception while notifying CE
Done
The location service is already running with pid: 16005
Done
Exception while notifying CE
```

```
Done
The configuration service is already running with pid: 16165
Done
Exception while notifying CE
Done
The matlabengine service is already running with pid: 1251
Done
Exception while notifying CE
Done
The nmsplb service is already running with pid: 1377
Done
Exception while notifying CE
```

cmxctl status

To view the status of one or all services, use the **cmxctl status** command.

cmxctl status [**analytics** | **agent** | **cache_6379** | **cache_6380** | **cache_6381** | **cassandra** | **configuration** | **confd** | **consul** | **database** | **haproxy** | **location** | **matlabengine** | **metrics** | **nmsplb** | **influxdb** | **iodocs** | **qlesspyworker**]

Syntax Description

analytics	Performs analytics on calculated location data.
agent	Manages CMX system lifecycle. starts, stops, and monitors all the services running in CMX.
cache_6379	Caches the service used by location service.
cache_6380	Caches the service used by analytics service.
cache_6381	Caches the service used by analytics service.
cassandra	Enables cassandra database service used by the location service for historical data.
configuration	Configures nodes and clusters.
confd	Internal service.
consul	Internal service.
database	Enables the database service used by analytics and configuration services.
haproxy	Enables the TCP or HTTP load balancer gateway to all service APIs.
location	Provides location service to compute location.
matlabengine	Provides access point heatmap for location service.
metrics	Collects system metrics.
nmsplb	Enables load balancer service used for distributing Network Mobility Services Protocol (NMSP) messages to location services.
influxdb	Enables database services used for storing statistics from various services.
iodocs	Enables online document service for REST API offered by various services.
qlesspyworker	Internal service.

Command Default

None

Examples

The following example shows how to display the status for the consul service:

```
[root@server]# cmxctl status consul

Done
The nodeagent service is currently running with PID: 16718
+--------------+---------------+---------+----------------+
| Host         | Service       | Status  | Uptime (HH:mm) |
+--------------+---------------+---------+----------------+
| cmx-master-1 | Analytics     | Running | 0 days, 05:17  |
+--------------+---------------+---------+----------------+
| cmx-master-1 | Cache_6379    | Running | 0 days, 05:17  |
+--------------+---------------+---------+----------------+
| cmx-master-1 | Cache_6380    | Running | 0 days, 05:17  |
+--------------+---------------+---------+----------------+
| cmx-master-1 | Cache_6381    | Running | 0 days, 05:17  |
+--------------+---------------+---------+----------------+
| cmx-master-1 | Cassandra     | Running | 0 days, 04:08  |
+--------------+---------------+---------+----------------+
| cmx-master-1 | Confd         | Running | 0 days, 03:40  |
+--------------+---------------+---------+----------------+
| cmx-master-1 | Configuration | Running | 0 days, 03:43  |
+--------------+---------------+---------+----------------+
| cmx-master-1 | Consul        | Running | 0 days, 05:17  |
+--------------+---------------+---------+----------------+
| cmx-master-1 | Database      | Running | 0 days, 05:17  |
+--------------+---------------+---------+----------------+
| cmx-master-1 | Haproxy       | Running | 0 days, 05:17  |
+--------------+---------------+---------+----------------+
| cmx-master-1 | Influxdb      | Running | 0 days, 05:17  |
+--------------+---------------+---------+----------------+
| cmx-master-1 | Iodocs        | Running | 0 days, 05:17  |
+--------------+---------------+---------+----------------+
| cmx-master-1 | Location      | Running | 0 days, 03:43  |
+--------------+---------------+---------+----------------+
| cmx-master-1 | Matlabengine  | Running | 0 days, 05:17  |
+--------------+---------------+---------+----------------+
| cmx-master-1 | Metrics       | Running | 0 days, 05:17  |
+--------------+---------------+---------+----------------+
| cmx-master-1 | Nmsplb        | Running | 0 days, 05:16  |
+--------------+---------------+---------+----------------+
| cmx-master-1 | Qlesspyworker | Running | 0 days, 05:23  |
+--------------+---------------+---------+----------------+
```

cmxctl stop

To shutdown a service, use the **cmxctl stop** command.

cmxctl stop {**analytics** | **agent** | **cache_6379** | **cache_6380** | **cache_6381** | **cassandra** | **configuration** | **confd** | **consul** | **database** | **haproxy** | **location** | **matlabengine** | **metrics** | **nmsplb** | **influxdb** | **iodocs** | **qlesspyworker**}

Syntax Description

analytics	Performs analytics on calculated location data.
agent	Manages CMX system lifecycle. starts, stops, and monitors all the services running in CMX.
cache_6379	Caches the service used by location service.
cache_6380	Caches the service used by analytics service.
cache_6381	Caches the service used by analytics service.
cassandra	Enables cassandra database service used by the location service for historical data.
configuration	Configures nodes and clusters.
confd	Internal service.
consul	Internal service.
database	Enables the database service used by analytics and configuration services.
haproxy	Enables the TCP or HTTP load balancer gateway to all service APIs.
location	Provides location service to compute location.
matlabengine	Provides access point heatmap for location service.
metrics	Collects system metrics.
nmsplb	Enables load balancer service used for distributing Network Mobility Services Protocol (NMSP) messages to location services.
influxdb	Enables database services used for storing statistics from various services.
iodocs	Enables online document service for REST API offered by various services.
qlesspyworker	Internal service.

Command Default

The services are running.

Examples

The following example shows how to stop the analytics service:

```
[root@server]# cmxctl stop analytics

Done
The nodeagent service is currently running with PID: 16987
Stopping analytics Process...
Service analytics with pid: 19095
Retrying..
Done
Successfully shutdown analytics Process.
```

cmxctl version

To know the Cisco Connected Mobile Experiences (Cisco CMX) RC version, use the **cmxctl version** command.

cmxctl version

Syntax Description

This command has no arguments or keywords.

Command Default

None

Examples

The following example shows how to display version information for Cisco CMX:

```
[root@server]# cmxctl version

Build Version   : 10.1.0-27
Build Time      : 2015-05-05 03:06:45.437430
----------------------------------------------------------------------
Name            : cmx-ng-container
Commit Count    : 17
Short Hash      : bf20ec1
----------------------------------------------------------------------
Name            : cmx-ng-location
Commit Count    : 5
Short Hash      : efc84fa
----------------------------------------------------------------------
Name            : cmx-ng-ui
Commit Count    : 5
Short Hash      : d793df7
----------------------------------------------------------------------
Name            : cmx-ova
Build Time      : Fri Feb 20 06:34:38 UTC 2015
----------------------------------------------------------------------
```

cmxctl config Commands

This section lists the **cmxctl config** commands:

- cmxctl config backup, page 1-16
- cmxctl config controllers, page 1-17
- cmxctl config import, page 1-18
- cmxctl config maps, page 1-19
- cmxctl config reload, page 1-20
- cmxctl config restore, page 1-21
- cmxctl config verify, page 1-22

cmxctl config backup

To back up a node, use the **cmxctl config backup** command.

cmxctl config backup

Syntax Description

This command has no arguments or keywords.

Command Default

None

Examples

The following example shows how to back up a node:

```
[root@server]# cmxctl config backup

Please enter the path for backup file [/tmp]:
[17:43:50] Preparing for backup...
[17:43:50] Backup Database...
[17:43:51] Backup Cache...
[17:43:51] Backup Cassandra...
[17:43:53] Backup InfluxDb...
[17:43:53] Backup Consul...
[17:43:53] Backup Floormaps...
[17:43:53] Backup node configuration...
[17:43:59] Creating tar file..
[17:43:59] Done Backup. Created backup file
/tmp/cmx_backup_cmx-vmdev117_2015_03_10_17_43.tar.gz
?]0;root@cmx-vmdev117:~[root@cmx-vmdev117 ~]# cmxctl config restore
?[?1034hPlease enter the backup file path:
/tmp/cmx_backup_cmx-vmdev117_2015_03_10_17_43.tar.gz
[17:44:12] Preparing for restore...
[17:44:12] Untarring backup file...
[17:44:13] Stopping all services...
[17:44:16] Restoring Database...
Restarting database...
[17:44:26] Restoring Cache...
Stopping cache_6379...
Restarting cache_6379...
Stopping cache_6381...
Restarting cache_6381...
Stopping cache_6380...
Restarting cache_6380...
[17:44:55] Restoring Cassandra...
Stopping Cassandra...
Restarting Cassandra...
..............
[17:45:19] Restoring Influxdb...
[17:45:19] Restoring consul...
[17:45:19] Restoring floormaps...
[17:45:19] Running Post Restore Tasks...
[17:45:19] Migrating Schemas...
[17:45:19] Migrating Cassandra schemas...
[17:45:20] Restarting all services...
[17:45:23] Done
```

cmxctl config controllers

To manage the Cisco Wireless Controllers (Cisco WLC), use the **cmxctl config controllers** command.

cmxctl config controllers {**add** | **delete** | **import** | **show**}

Syntax Description

add	Adds a Cisco WLC.
delete	Deletes a Cisco WLC.
import	Imports a Cisco WLC from PI by providing PI credentials or by placing an exported PI MAP file in the /opt directory of the CMX server and providing the path to exported map file.
show	Shows information pertaining to a Cisco WLC.

Command Default

None

Usage Guidelines

After a controller is added, the message “controller added successfully” may be shown. Note that this refers only to the correct parsing of the command. You should issue a **cmxctl controllers show** command to ensure that the controller is not active.

Examples

The following example shows how to display the Cisco WLC information:

```
[root@server]# cmxctl config controllers show

+--------------+------+------------+----------------+------+--------+
| IP Address | Type | Version | Device Version | SHA2 | Status |
+--------------+------+------------+----------------+------+--------+
| 30.30.30.65 | WLC | 8.0.72.141 | - | No | ACTIVE |
+--------------+------+------------+----------------+------+--------+
| 30.30.30.44 | WLC | 8.0.72.141 | - | No | ACTIVE |
+--------------+------+------------+----------------+------+--------+
| 30.30.30.46 | WLC | 8.0.72.141 | - | No | ACTIVE |
+--------------+------+------------+----------------+------+--------+
| 30.30.30.70 | WLC | 8.0.72.141 | - | No | ACTIVE |
+--------------+------+------------+----------------+------+--------+
| 30.30.30.93 | WLC | 8.0.72.141 | - | No | ACTIVE |
+--------------+------+------------+----------------+------+--------+
| 30.30.30.97 | WLC | 8.0.72.141 | - | No | ACTIVE |
+--------------+------+------------+----------------+------+--------+
| 30.30.30.35 | WLC | 8.0.72.141 | - | No | ACTIVE |
+--------------+------+------------+----------------+------+--------+
| 30.30.30.58 | WLC | 8.0.72.141 | - | No | ACTIVE |
+--------------+------+------------+----------------+------+--------+
| 30.30.30.82 | WLC | 8.0.72.141 | - | No | ACTIVE |
+--------------+------+------------+----------------+------+--------+
| 30.30.30.84 | WLC | 8.0.72.141 | - | No | ACTIVE |
+--------------+------+------------+----------------+------+--------+
| 30.30.30.53 | WLC | 8.0.72.141 | - | No | ACTIVE |
+--------------+------+------------+----------------+------+--------+
```

cmxctl config import

To imports maps and Cisco Wireless Controllers (Cisco WLC) from Cisco Prime Infrastructure, use the **cmxctl config import** command.

cmxctl config import {prime | status}

Syntax Description

prime	Imports maps from the Cisco Prime Infrastructure
status	Shows import status.

Command Default

None

Examples

The following example shows how to import maps and Cisco WLC from Prime Infrastructure:

```
[root@server]# cmxctl config import prime

Please enter PI ip address: x.x.x.x
Please enter PI username [root]: root
Please enter PI password [Public123]:

Import successfully started from PI x.x.x.x. Check import status using cmxctl config
import status.
```

cmxctl config maps

To import and manage maps, use the **cmxctl config maps** command.

cmxctl config maps {address | delete | import}

Syntax Description

address	Imports addresses for the maps.
delete	Deletes the campus map.
import	Imports map from the Cisco Prime Infrastructure.

Command Default

None.

Examples

The following example shows how to import and manage maps:

```
[root@server]# cmxctl config maps import

Please specify import type [PI / FILE] [FILE]: PI
Please enter PI ip address: x.x.x.x
Please enter PI username [root]: root
Please enter PI password [Public123]:

Import successfully started from PI 173.37.206.3. Check import status using cmxctl config
import status.
```

cmxctl config reload

To force the generation of config files, use the **cmxctl config reload** command.

cmxctl config reload

Syntax Description

This command has no arguments or keywords.

Command Default

None

Examples

The following example shows how to forcefully generate configuration files:

```
[root@server]# cmxctl config reload

2015-03-10T17:45:50Z cmx-vmdev117 -verbose[17174]: WARNING Skipping confd config file.
2015-03-10T17:45:50Z cmx-vmdev117 -verbose[17174]: INFO Target config
/opt/cmx/etc/analytics.conf in sync
2015-03-10T17:45:50Z cmx-vmdev117 -verbose[17174]: INFO Target config
/opt/cmx/etc/cassandra/cassandra-env.sh in sync
2015-03-10T17:45:50Z cmx-vmdev117 -verbose[17174]: INFO Target config
/opt/cmx/etc/cassandra/cassandra.yaml in sync
2015-03-10T17:45:50Z cmx-vmdev117 -verbose[17174]: INFO Target config
/opt/cmx/etc/collectd.conf in sync
2015-03-10T17:45:50Z cmx-vmdev117 -verbose[17174]: INFO Target config
/opt/cmx/etc/configuration.conf in sync
2015-03-10T17:45:50Z cmx-vmdev117 -verbose[17174]: INFO Target config
/opt/cmx/etc/connect.conf in sync
2015-03-10T17:45:50Z cmx-vmdev117 -verbose[17174]: INFO Target config
/opt/cmx/etc/halo.conf in sync
2015-03-10T17:45:50Z cmx-vmdev117 -verbose[17174]: INFO Target config
/opt/cmx/etc/haproxy.cfg in sync
2015-03-10T17:45:50Z cmx-vmdev117 -verbose[17174]: INFO Target config
/opt/cmx/etc/influxdb.conf in sync
2015-03-10T17:45:50Z cmx-vmdev117 -verbose[17174]: INFO Target config
/opt/cmx/etc/location.conf in sync
2015-03-10T17:45:50Z cmx-vmdev117 -verbose[17174]: INFO Target config
/opt/cmx/etc/matlabengine.conf in sync
2015-03-10T17:45:50Z cmx-vmdev117 -verbose[17174]: INFO Target config
/opt/cmx/etc/nmsplb.conf in sync
2015-03-10T17:45:50Z cmx-vmdev117 -verbose[17174]: INFO Target config
/opt/cmx/etc/nmspproxy.conf in sync
2015-03-10T17:45:50Z cmx-vmdev117 -verbose[17174]: INFO Target config
/opt/cmx/etc/postgresql.conf in sync
2015-03-10T17:45:50Z cmx-vmdev117 -verbose[17174]: INFO Target config
/opt/cmx/etc/redis_6379.conf in sync
2015-03-10T17:45:50Z cmx-vmdev117 -verbose[17174]: INFO Target config
/opt/cmx/etc/redis_6380.conf in sync
2015-03-10T17:45:50Z cmx-vmdev117 -verbose[17174]: INFO Target config
/opt/cmx/etc/redis_6381.conf in sync
2015-03-10T17:45:50Z cmx-vmdev117 -verbose[17174]: ERROR template:
redis.template.conf:15:20: executing "redis.template.conf" at <getv ($tag | printf ...>:
error calling getv: key does not exist
```

cmxctl config restore

To restore a node, use the **cmxctl config restore** command.

cmxctl config restore

Syntax Description

This command has no arguments or keywords.

Command Default

None.

Examples

The following example shows how to restore a node:

```
[root@server]# cmxctl config restore

Please enter the backup file path: /tmp/cmx_backup_cmx-vmdev117_2015_03_10_17_43.tar.gz
[17:44:12] Preparing for restore...
[17:44:12] Untarring backup file...
[17:44:13] Stopping all services...
[17:44:16] Restoring Database...
Restarting database...
[17:44:26] Restoring Cache...
Stopping cache_6379...
Restarting cache_6379...
Stopping cache_6381...
Restarting cache_6381...
Stopping cache_6380...
Restarting cache_6380...
[17:44:55] Restoring Cassandra...
Stopping Cassandra...
Restarting Cassandra...
...............
[17:45:19] Restoring Influxdb...
[17:45:19] Restoring consul...
[17:45:19] Restoring floormaps...
[17:45:19] Running Post Restore Tasks...
[17:45:19] Migrating Schemas...
[17:45:19] Migrating Cassandra schemas...
[17:45:20] Restarting all services...
[17:45:23] Done
```

cmxctl config verify

To verify the Cisco CMX installation and configuration, use the **cmxctl config verify** command.

cmxctl config verify

Syntax Description

This command has no arguments or keywords.

Command Default

None

Examples

The following example shows how to verify the Cisco CMX installation and configuration:

```
[root@server]# cmxctl config verify

Verifying node configuration...
NetworkManager: unrecognized service
Consul v0.4.1
Consul Protocol: 2 (Understands back to: 1)
confd 0.6.0
+-----------------+--------------------------------+------------+-------------------------
--+
|     module      |             check              |   passed   |            msg
|
+=================+================================+============+=========================
==+
| netman_stopped  | NetworkManager service is not  | Success    |
|
|                 | running                        |            |
|
+-----------------+--------------------------------+------------+-------------------------
--+
| matlabengine    | http://matlabengine.service.co | Failed     | check the log files
under |
|                | nsul:5577/api/services/matlabe |            | /opt/cmx/var/log
|
|                 | ngine/status                   |            |
|
+-----------------+--------------------------------+------------+-------------------------
--+
| database        | connect to database port:5432  | Success    |
|
+-----------------+--------------------------------+------------+-------------------------
--+
| consul_dns      | 127.0.0.1 (consul) is present  | Success    |
|
|                 | as dns server in               |            |
|
|                 | /etc/resolv.conf               |            |
|
+-----------------+--------------------------------+------------+-------------------------
--+
| etchost_hacks   | consul service hostnames not   | Success    |
|
|                 | static in /etc/hosts           |            |
|
+-----------------+--------------------------------+------------+-------------------------
--+
| analytics       | http://analytics.service.consu | Failed     | check the log files
under |
|                 | l:5556/api/services/analytics/ |            | /opt/cmx/var/log
|
|                 | status                         |            |
|
+-----------------+--------------------------------+------------+-------------------------
--+
| hostname_ping   | ping to hostname:cmx-master-1  | Success    |
|
+-----------------+--------------------------------+------------+-------------------------
--+
| location        | http://location.service.consul | Failed     | check the log files
under |
|                 | :5555/api/services/location/st |            | /opt/cmx/var/log
|
|                 | atus                           |            |
|
+-----------------+--------------------------------+------------+-------------------------
--+
| confd_installed | Confd is installed             | Success    |
|
+-----------------+--------------------------------+------------+-------------------------
--+
| consul_installe | Consul is installed            | Success    |
|
| d               |                                |            |
|
+-----------------+--------------------------------+------------+-------------------------
--+
| nmsplb          | http://nmsplb.service.consul:6 | Failed     | check the log files
under |
|                 | 001/api/services/nmsplb/status |            | /opt/cmx/var/log
|
+-----------------+--------------------------------+------------+-------------------------
--+
| configuration   | http://configuration.service.c | Failed     | check the log files
under |
|                 | onsul:6000/api/services/config |            | /opt/cmx/var/log
|
|                 | uration/status                 |            |
|
+-----------------+--------------------------------+------------+-------------------------
--+
| cassandra       | connect to cassandra port:9042 | Success    |
|
+-----------------+--------------------------------+------------+-------------------------
--+-+
```