

show



# Project Olympus Rack Manager

Software: Command-Line Interface

**Author:** Microsoft  
**Document:**  
**Revision:** DRAFT  
**Version:** V.0.10.5  
**Date:** October 28, 2020

Distribution:

- ☐ Internal Only
- ☒ External All
- ☐ External Restricted

If restricted, specify restricted to whom:

---

---

(c) 2015 Microsoft Corporation. All rights reserved.

This document is provided "AS-IS" with no warranties. Information and views expressed in this document may change without notice. You bear the risk of using it.

Microsoft and/or third parties may have intellectual property rights covering the subject matter in this document. Except as may be expressly provided in a separate agreement, if any, the furnishing of this document does not grant any licenses to any such intellectual property rights. Instead, this document is for your internal reference purposes. This document and its contents are confidential and proprietary to Microsoft. It is disclosed and can be used only pursuant to a non-disclosure agreement between you and Microsoft.

CONFIDENTIAL

## Revision History

Revision/ Date	Notes
V0.1.0	<ul style="list-style-type: none"> <li>Separated specification into three parts: hardware, software, and server implementations</li> </ul>
V0.1.1	<ul style="list-style-type: none"> <li>Added CLI commands</li> </ul>
V0.2.5	<ul style="list-style-type: none"> <li>Update command format, and specification numbering to align milestone release</li> </ul>
V0.3	<ul style="list-style-type: none"> <li>Several new feature additions <ul style="list-style-type: none"> <li>TFTP List files, Get, Put, Delete file</li> <li>Remote media mount, unmount</li> <li>RM ssh sessions list, kill</li> <li>MTE command file inputs from TFTP location (fixed names)</li> <li>MTE command bug fixes</li> </ul> </li> </ul>
V0.4	<ul style="list-style-type: none"> <li>New feature additions <ul style="list-style-type: none"> <li>BIOS firmware update</li> <li>Manager recovery menu</li> <li>Manager fwupdate</li> <li>UPS feature</li> <li>Tab completion feature</li> </ul> </li> </ul>
V0.5	<ul style="list-style-type: none"> <li>Add MTE lock and firewall commands.</li> <li>Fixed typos in mte commands (Imbench3 -&gt; Imbench)</li> <li>Throttlecontrol/throttlebypass argument description</li> </ul>
V0.6	<ul style="list-style-type: none"> <li>Several new feature additions <ul style="list-style-type: none"> <li>Datasafe (NVDIMM, PCIe) settings</li> <li>System file transfer operations</li> <li>Port Pairing</li> <li>Manager JTAG service and system FPGA JTAG service</li> <li>ITP service</li> </ul> </li> </ul>
V0.7	<ul style="list-style-type: none"> <li>Add Force throttle, throttle count get/set commands</li> <li>Add voltage, current readings to 'powermeter reading' command</li> <li>Add powermeter calibration get/set commands</li> <li>Add restore defaults command</li> <li>Add ping and traceroute commands</li> <li>Add sftp commands</li> <li>Remove duplicate tftp get command</li> <li>Add 'tfb' command to mte command list to provision TFBs</li> </ul>
V0.8	<ul style="list-style-type: none"> <li>NVDIMM save as part of 'port off' and save status as part of 'port status' command</li> <li>Update led name to UID for the system LED commands</li> </ul>

V0.9	<ul style="list-style-type: none"><li>Power commands (port state/on/off, system state/on/off/reset) description altered to account for NVDIMM/datasave operation scenarios.</li></ul>
V0.10	<ul style="list-style-type: none"><li>New feature additions<ul style="list-style-type: none"><li>show manager port inventory</li><li>Extend the data collection for telemetry service (to include server inlet,outlet,cpu0,cpu1 and temperatures)</li><li>New work flow for PSU Update</li></ul></li></ul>
V0.10.1	<ul style="list-style-type: none"><li>Add PubSub commands(Get/Set)</li></ul>
V0.10.2	<ul style="list-style-type: none"><li>Add Soc supported commands</li></ul>
V0.10.3	<ul style="list-style-type: none"><li>Add BMC and CPLD firmware update and query</li></ul>
V0.10.4	<ul style="list-style-type: none"><li>Added commands pubsub, userconfigs, serial terminal stats and check</li></ul>
V0.10.5	<ul style="list-style-type: none"><li>Added multi-policy power related commands</li></ul>

Contents

1	Overview of Rack Manager Command-Line Interface Specification .....	11
2	Command Line Interface .....	11
3	User Roles and API Access .....	12
3.1	Encryption and Service Credentials .....	12
3.2	Client Credentials/Authentication .....	12
3.3	Role-Based API-Level Authorization .....	12
4	Rack Manager Service .....	14
5	Network Commands.....	15
5.1	Show Network .....	15
5.2	Show Network interface.....	15
5.3	Show Network Ping .....	16
5.4	Show Network Traceroute.....	17
5.5	Set Network Interface Static (Set Static) .....	17
5.6	Set Network Interface Dhcp (Set DHCP) .....	18

5.7    *Set Network Interface Enable*.....18

5.8    *Set Network Interface Disable*.....19

5.9    *Show Network route* .....19

5.10    *Set Network route* .....20

**6    Manager Commands ..... 21**

6.1    *Show Manager Info*.....23

6.2    *Show Manager Health* .....24

6.3    *Show Manager Version* .....25

6.4    *Show Manager Relay* .....25

6.5    *Show Manager LED* .....25

6.6    *Show Manager Port State* .....26

6.7    *Show Manager Port Presence* .....27

6.8    *Show Manager Port Pairing* .....27

6.9    *Show Manager Port Inventory* .....27

6.10    *Show Manager Log* .....28

6.11    *Show Manager Inventory*.....29

6.12    *Show Manager scandevice*.....29

6.13    *Show Manager Fru* .....29

6.14    *Show Manager tftp Status* .....30

6.15    *Show Manager NFS Status*.....30

6.16    *Show PowerMeter Alert* .....31

6.17    *Show PowerMeter Limit*.....31

6.18    *Show Manager PowerMeter Reading* .....32

6.19    *Show Manager PowerMeter Multipolicy Active*.....32

6.20    *Show Manager PowerMeter Multipolicy Single*.....33

6.21    *Show Manager PowerMeter Multipolicy All* .....34

6.22    *Show Manager Power Meter Version* .....35

6.23    *Show Manager Power Reading* .....35

6.24    *Show Manager Power Status* .....35

6.25    *Show Manager NTP Status*.....36

6.26    *Show Manager NTP Server*.....36

6.27    *Show Manager ITP Status* .....37

SW: RM CLI

6.28 *Show manager throttle local*.....37

6.29 *Show manager throttle count* .....38

6.30 *Show manager throttle row* .....38

6.31 *Show manager recovery (boot strap)*.....39

6.32 *Show Manager Session list*.....39

6.33 *Show Manager Tftp list* .....40

6.34 *Show Manager Time* .....41

6.35 *Show Manager Type*.....41

6.36 *Show Manager menu* .....42

6.37 *Show Manager fwupdate status*.....42

6.38 *Show Manager fwupdate list* .....42

6.39 *Show Manager jtag status* .....43

6.40 *Show manager powermeter calibration* .....43

6.41 *Show manager throttle count* .....44

6.42 *Show manager pubsub config*.....44

6.43 *Show manager userconfigs*.....45

6.44 *how manager serialstats*.....46

6.45 *Show manager serialcheck*.....46

6.46 *Show manager ssh status*.....47

6.47 *Set Manager LED On* .....50

6.48 *Set Manager LED Off*.....50

6.49 *Set Manager Relay On*.....50

6.50 *Set Manager Relay Off* .....50

6.51 *Set Manager Port On*.....51

6.52 *Set Manager Port Off* .....52

6.53 *Set Manager Port pairing add*.....53

6.54 *Set Manager Port pairing remove*.....54

6.55 *Set Manager Log Clear*.....54

6.56 *Set Manager Fru*.....55

6.57 *Set Manager ClearFaults*.....55

6.58 *Set Manager PowerMeter Alert* .....56

6.59 *Set Manager PowerMeter Limit*.....56

6.60 Set Manager PowerMeter Multipolicy .....56

6.61 Set Manager PowerMeter ClearMax .....57

6.62 Set Manager PowerMeter ClearFaults .....58

6.63 Set Manager TFTP Start .....58

6.64 Set Manager TFTP Stop .....58

6.65 Set Manager NFS Start.....59

6.66 Set Manager NFS Stop.....59

6.67 Set Manager NTP Start.....60

6.68 Set Manager NTP Stop .....60

6.69 Set Manager NTP Restart.....60

6.70 Set Manager NTP Enable.....61

6.71 Set Manager NTP Disable.....61

6.72 Set Manager NTP Server .....61

6.73 Set Manager ITP Start .....62

6.74 Set Manager ITP Stop.....62

6.75 Set Manager ITP Restart .....62

6.76 Set Manager ITP Enable .....63

6.77 Set mSet Manager ITP Disable .....63

6.78 Set Manager jtag Start.....64

6.79 Set Manager jtag Stop .....64

6.80 Set Manager jtag Restart .....64

6.81 Set Manager jtag Enable.....65

6.82 Set Manager jtag Disable.....65

6.83 Set Manager Hostname .....66

6.84 Set Manager throttle local .....66

6.85 Set Manager throttle count.....66

6.86 Set Manager throttle force.....67

6.87 Set Manager throttle row .....67

6.88 Set Manager Recovery On.....68

6.89 Set Manager Recovery Off .....69

6.90 Set Manager Session Kill .....69

6.91 Set Manager Tftp Delete .....70

SW: RM CLI

6.92 Set Manager Tftp/Sftp Get ..... 70

6.93 Set Manager Tftp/Sftp Put ..... 71

6.94 Set Manager Time ..... 72

6.95 Set Manager menu On ..... 72

6.96 Set Manager menu Off ..... 73

6.97 Set Manager fwupdate ..... 73

6.98 Set manager defaults ..... 74

6.99 Show user info ..... 74

6.100 Set User add ..... 75

6.101 Set user update ..... 75

6.102 Set User Delete ..... 76

6.103 Set manager powermeter calibration ..... 76

6.104 Set manager throttle force enable ..... 77

6.105 Set manager throttle count clear ..... 77

6.106 Set manager pubsub config ..... 78

6.107 Set manager userconfigs ..... 78

6.108 Set manager ssh config ..... 79

**7 Server Commands ..... 80**

7.1 Show System Info ..... 82

7.2 Show system fru ..... 83

7.3 Show system State ..... 83

7.4 Show System Presence ..... 85

7.5 Show System Default Power ..... 85

7.6 Show System Power Limit ..... 85

7.7 Show System Power Reading ..... 86

7.8 Show System Power Alert ..... 86

7.9 Show System Power Throttle ..... 87

7.10 Show System FPGA Health ..... 87

7.11 Show System FPGA Mode ..... 88

7.12 Show System FPGA Temp ..... 88

7.13 Show System FPGA I2CVersion ..... 89

7.14 Show System FPGA AssetInfo ..... 89

SW: RM CLI

7.15	Show System Boot.....	90
7.16	Show system Log read.....	90
7.17	Show Bios Config .....	91
7.18	Show BIOS Code .....	92
7.19	Show system bios update.....	93
7.20	Show system bmc update.....	93
7.21	Show system cpld update.....	93
7.22	Show Tpm Physical Presence.....	94
7.23	Show System Psu Info.....	94
7.24	Show system psu battery .....	95
7.25	Show system psu update.....	96
7.26	Show system psu version.....	96
7.27	Show system nvme .....	97
7.28	Show System Type.....	98
7.29	Show System datasafe policy .....	98
7.30	Show System datasafe trigger.....	98
7.31	Show System led.....	99
7.32	Show System File List.....	99
7.33	Set Log Clear.....	100
7.34	Set Power Alert.....	100
7.35	Set System On.....	101
7.36	Set System Off .....	102
7.37	Set System Reset .....	103
7.38	Set System LED On.....	104
7.39	Set System LED Off .....	104
7.40	Set System Default Power State On .....	105
7.41	Set System Default Power State Off .....	105
7.42	Set System Power Limit Value .....	106
7.43	Set System Power Limit On.....	107
7.44	Set System Power Limit Off .....	107
7.45	Set system Boot.....	108
7.46	Set System Bios Config .....	109



7.47    *Set system psu clear* .....109

7.48    *Set system psu battery* .....110

7.49    *Set system psu update*.....110

7.50    *Set system bios update*.....111

7.51    *Set system bmc update* .....111

7.52    *Set system cpld update*.....112

7.53    *Set FPGA Bypass Enable* .....112

7.54    *Set FPGA Bypass Disable* .....113

7.55    *Set Tpm Physical Presence* .....113

7.56    *Set System Rmedia Mount* .....113

7.57    *Set System Rmedia Unmount*.....114

7.58    *Set System file delete* .....115

7.59    *Set System file get* .....115

7.60    *Set System file put* .....116

7.61    *Start Serial Session* .....116

7.62    *Stop Serial Session*.....117

7.63    *Start JTAG service*.....117

7.64    *Stop JTAG service* .....118

**8    Switch Commands**..... **119**

8.1    *Show Switch Info* .....119

8.2    *Show Switch Port*.....119

8.3    *Set Switch Config*.....120

8.4    *Set Switch Upgrade* .....120

8.5    *Set Switch Reset* .....120

8.6    *Start Auxiliary Console* .....121

**9    UPS Commands** ..... **122**

9.1    *Show ups voltage* .....122

9.2    *Show ups power* .....122

9.3    *Show ups current*.....122

9.4    *Show ups state* .....123

9.5    *Show ups test* .....123

9.6    *Show ups config* .....123

SW: RM CLI

- 9.7    *Show ups timer*.....124
- 9.8    *Show ups info* .....124
- 9.9    *Show ups alarm*.....124
  
- 10    Help Commands ..... 125**
  - 10.1    *Help* .....125
  - 10.2    *help show* .....125
  - 10.3    *help set*.....126
  - 10.4    *help start* .....126
  - 10.5    *help stop*.....127
  - 10.6    *help user*.....128
  
- 11    MTE Commands ..... 129**
  
- 12    Appendix: Commonly Used of Acronyms..... 132**

SW: RM CLI

# 1 Overview of Rack Manager Command-Line Interface Specification

This document, “Project Olympus - Software: Rack Manager Command-Line Interface,” describes the command-line interface (CLI) that can be used to manage the Project Olympus systems. Systems management in the OCS system is designed to present a consistent, optimized interface. A Rack Manager (RM) provides the front end through an applications interface (RESTful web API) for automated management and the command-line interface for manual management. It manages all devices within the rack and communicates directly with the server management system through network.

## 2 Command Line Interface

The command line interface is accessible over Serial (UART Interface) and SSH (Ethernet Interface). Once a user connects to the Serial or SSH interfaces they will be presented with a logon prompt. After logging into the console, the following section describes the commands a user can execute.

## 3 User Roles and API Access

The OCS system Rack Manager web service provides a full set of security features, including encryption, integrity, authentication, and fine-grained API-level authorization.

### 3.1 Encryption and Service Credentials

All data communication between the Rack Manager web service and clients (web browser or command-line interface) is encrypted using secure socket layers (SSL). OCS uses signature-based checksum (signed packets) to prevent tampering and sends data secure HTTP (HTTPS) to ensure integrity. The OCS Rack Manager web service is also authenticated against the client using Microsoft Certificate Services.

### 3.2 Client Credentials/Authentication

Client authentication is based on machine-local accounts and certificates. Through CLI, client credentials are automatically obtained from the logged-in user.

### 3.3 Role-Based API-Level Authorization

Client authorization to the Rack Manager web service is provided at the granularity of the service APIs. The Rack Manager APIs are categorized into three security domains:

- U1—APIs that perform Rack Manager management functions and manage devices that are connected to the Rack Manager (for example, blades and power supply units).
- U2—APIs that manage devices (for example, blades and power supply units) that are connected to the Rack Manager.
- U3—APIs that perform only read-only operations.

Note that U1 includes all OCS Rack Manager APIs, while U2 and U3 include only a subset of the OCS Rack Manager APIs. Note also that U3 is a subset of U2, which is a subset of U1.

Users authorized to perform Rack Manager functions can be categorized into three groups:

- admin—Users in this group are authorized to perform functions in U1. They have access to all APIs, including APIs for Rack Manager management functions like “firmware update”, “networking”, “add-user” and rack power cap etc.
- operator—Users in this group are authorized to perform functions in U2, and have access to all APIs except those for Rack Manager management functions.
- user—Users in this group are authorized to perform functions in U3, and can only access read-only APIs.

Any user attempting to access the Rack Manager APIs will be authorized based on their role or group. Users who do not belong to any of the three authorization roles or groups are denied access to Rack Manager API functionalities.

The three groups are available in each Rack Manager, and local Rack Manager users can be assigned to any of the three local roles (admin, operator, user) using the “add user” and “change user” APIs.

## 4 Rack Manager Service

- [Show Version](#)

**Description:**

This command shows the Manager version.

**Syntax:**

```
wcscli# show/sh version
```

**Sample usage:**

```
wcscli# show version
```

**Sample output:**

```
*****
Microsoft OCS Rack Manager software version summary

Package version   : 1.0.3.77

Rootfs version    : 1.0.3.77

U-Boot version    : 1.0.3.77

Kernel version    : 1.0.3.77

Devicetree version : 1.0.3.77

PRU FW version    : 0.7
*****
```

SW: RM CLI

## 5 Network Commands

- [Show Network](#)
- [Show network interface](#)
- [Show Network route](#)
- [Set Network Static](#)
- [Set Network DHCP](#)
- [Set Network Interface enable](#)
- [Set Network Interface disable](#)
- [Set Network route](#)

---

### 5.1 Show Network

**Description:**

This command shows the Manager physical interfaces list.

**Syntax:**

```
Wcscli# show/sh network
```

**Sample usage:**

To get all the network configuration details for the manager, use the following command:

```
wcscli# show network
```

**Sample output:**

```
wcscli# sh network
```

```
Interfaces_count: 2
```

```
Interfaces_list:
```

```
1: eth1
```

```
2: eth0
```

```
Completion Code: Success
```

Returns all the available interfaces

---

### 5.2 Show Network interface

**Description:**

This command shows the Rack Manager or Row Manager specific interface details.

**Syntax:**

```
Wcscli# show/sh network interafce -i <interface>{eth0,eth1}
```

**Sample usage:**

To get the network configuration details for the rack or row, use the following command:

```
wcscli# show network interface -i <eth0 or eth1>
```

**Sample output:**

```
wcscli# sh network interface -i eth0
IPv4Addresses:
  Address: 172.16.138.221
  AddressOrigin: DHCP
  Gateway: 172.16.0.1
  SubnetMask: 255.255.0.0
IPv6Addresses:
  Address: fe80::be83:85ff:fefa:6095
  AddressOrigin:
  AddressState: True
  PrefixLength: 64
Id: eth0
MACAddress: BC:83:85:FA:60:95
Status_Health: Ok
Status_State: Up
Completion Code: Success
```

### 5.3 Show Network Ping

**Description:**

This command will test network connectivity to another host on the network.

**Syntax:**

```
Wcscli# show/sh network ping [-c <count>] [-w <timeout>] -a <address>
```

-c → Stop pinging after this many requests have been sent.

-w → Timeout is seconds to wait for the first response to be received.

-a → The IP address of the host to ping.

**Sample usage:**

```
Wcscli# show network ping -a 192.168.1.1
```

**Sample output:**

```
PING 192.168.1.1 (192.168.1.1): 56 data bytes
64 bytes from 192.168.1.1: seq=0 ttl=64 time=0.476 ms
64 bytes from 192.168.1.1: seq=1 ttl=64 time=0.336 ms
64 bytes from 192.168.1.1: seq=2 ttl=64 time=0.413 ms
```



---

```
64 bytes from 192.168.1.1: seq=3 ttl=64 time=0.342 ms
--- 192.168.1.1 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 0.336/0.391/0.476 ms
```

**Notes:**

Use CTRL+C to stop the ping command.

## 5.4 Show Network Traceroute

**Description:**

Discovers the route to a remote host on the network.

**Syntax:**

```
Wcscli# show/sh network traceroute [-u] -a <address>
```

-u → Use UDP packets instead of ICMP packets to discover the route.

-a → The IP address of the host to trace.

**Sample usage:**

```
Wcscli# show network traceroute -a 192.168.1.1
```

**Sample output:**

```
traceroute to 192.168.1.1 (192.168.1.1), 30 hops max, 38 byte packets
 1 192.168.1.1 (192.168.1.1) 0.938 ms 0.265 ms 0.136 ms
```

**Notes:**

In order for UDP traceroute to work, the firewall on the rack manager must be disabled.

---

## 5.5 Set Network Interface Static (Set Static)

**Description:**

This command sets static network properties to the Manager.

**Syntax:**

```
Wcscli# set network static -i <interface name> <-a IP addr> -s
<subnetmask> -g {gateway}
```

-i → interface name {eth0 or eth1}

-a → IP address (Required for Static IP)>

-s → subnetmask (Required for subnet mask)>

-g → gateway

[-h]

-h - help; display the correct syntax

---

**Sample usage:**

To set the network configuration for the rack, use the following command:

```
Wsccli# set network static -i eth1 -a 192.168.1.12 -s 255.255.255.0 -g 192.168.1.1
```

**Sample output:**

Completion Code : Success

The command will execute successfully, and there will be intermittent connection loss to manager because of the network config update.

---

## 5.6 Set Network Interface Dhcp (Set DHCP)

**Description:**

This command sets dhcp to the Manager.

**Syntax:**

```
Wsccli# set network dhcp -i <interface name>
```

-i → interface name {eth0 or eth1}

[-h]

-h - help; display the correct syntax

**Sample usage:**

To set the network configuration for the manager, use the following command:

```
Wsccli# set network dhcp -i eth0
```

**Sample output:**

Completion Code : Success

The command will execute successfully, and there will be intermittent connection loss to rack manager because of the network config. update.

---

## 5.7 Set Network Interface Enable

**Description:**

This command sets disable interface

**Syntax:**

```
Wsccli# set network enable -i <interface name>
```

---

```
-i → interfacse name {eth0 or eth1}
[-h]
    -h - help; display the correct syntax
```

**Sample usage:**

To set the network configuration for the manager, use the following command:

```
Wsccli# set network enable -i eth0
```

**Sample output:** Completion Code : Success

*The command will execute successfully, and there will be intermittent connection loss to manager because of the network cofig. update.*

---

## 5.8 Set Network Interface Disable

**Description:**

This command sets enable interface

**Syntax:**

```
Wsccli# set network disable -i <interface name>
```

```
-i → interfacse name {eth0 or eth1}
[-h]
    -h - help; display the correct syntax
```

**Sample usage:**

To set the network configuration for the manager, use the following command:

```
Wsccli# set network disable -i eth0
```

**Sample output:**

Completion Code : Success

The command will execute successfully, and there will be intermittent connection loss to manager

---

## 5.9 Show Network route

**Description:**

- Display the current management network and gateway.

**Syntax:**

```
Wsccli# show network route
[-h]
    -h - help; display the correct syntax
```

---

**Sample usage:**

To set the network configuration for the manager, use the following command:

```
Wcscli# show network route
```

**Sample output:**

Completion Code : Success

---

## 5.10 Set Network route

**Description:**

Set the static route for accessing the management network

**Syntax:**

```
Wcscli# set network route -g <gateway> -n <subnet>
```

-g → Gateway is the IP address of the switch, which is the gateway for that network.

-n → Subnet is the netmask of the management network. It can be represented as either a dotted quad (255.255.255.0) or the number of network bits (24).

[-h]

-h - help; display the correct syntax

**Sample usage:**

To set the network configuration for the manager, use the following command:

```
Wcscli# set network route -g <ip-address> -n <255.255.255.0>
```

**Sample output:**

Completion Code : Success

## 6 Manager Commands

### *Show Commands*

- [show manager info](#)
- [show manager health](#)
- [Show manager Version](#)
- [show manager relay](#)
- [show manager led](#)
- [Show manager port state](#)
- [Show manager port presence](#)
- [Show manager port pairing](#)
- [Show manager port inventory](#)
- [Show manager log](#)
- [Show manager inventory](#)
- [Show manager scandevice](#)
- [show manager fru](#)
- [show manager tftp status](#)
- [show manager nfs tatus](#)
- [show manager powermeter alert](#)
- [show manager powermeter limit](#)
- [show manager powermeter reading](#)
- [show manager powermeter multipolicy active](#)
- [show manager powermeter multipolicy single](#)
- [show manager powermeter multipolicy all](#)
- [show manager powermeter version](#)
- [show manager power reading](#)
- [show manager power status](#)
- [show manager ntp status](#)
- [show manager ntp server](#)
- [show manager itp status](#)
- [show manager throttle local](#)
- [show manager throttle count](#)
- [show manager throttle row](#) (ROW manager command)
- [show manager recovery](#) (boot strap : Row Manager Command)
- [show manager session list](#)
- [show manager tftp list](#)
- [Show manager time](#)
- [Show manager Type](#)
- [Show manager menu](#)

- [Show manager fwupdate status](#)
- [Show manager fwupdate list](#)
- [Show manager jtag status](#)
- [Show manager powermeter calibration](#)
- [Show manager throttle row](#)
- [Show manager throttle row](#)
- [Show manager recovery \(boot strap\)](#)
- [Show Manager Session list](#)
- [Show Manager Tftp list](#)
- [Show Manager Time](#)
- [Show Manager Type](#)
- [Show Manager menu](#)
- [Show Manager fwupdate status](#)
- [Show Manager fwupdate list](#)
- [Show Manager jtag status](#)
- [Show manager powermeter calibration](#)
- [Show manager pubsub config](#)
- [show manager userconfigs](#)
- [show manager serialstats](#)
- [show manager serialcheck](#)
- [show manager ssh status](#)

## 6.1 Show Manager Info

### Description:

This command gets status information about manager components including the following:

- **Servers**
  - o GUID, presence, powerstate etc.,
- **Power supplies**
  - o Information (for example, power draw, input\_voltage etc.,) Hot Swap Controllers (HSC) that manage power to manager equipment
- **Manager**
  - o MAC/IP address of the network interfaces, and version information etc.,

### Syntax:

```
Wcscli# show manager info [-s] [-p] [-m] [-h]
```

- s – Show information about servers
- p – Show information about power supplies
- m – Show manager information
- h – Help, display the correct syntax

### Sample usage:

```
wcscli# show manager info → Returns all manager info  
wcscli# show manager info -s → Returns only server details  
wcscli# show manager info -p → Returns only power supply details  
wcscli# show manager info -m → Returns only manager details
```

### Sample output:

```
wcscli# sh manager info -m  
Asset Info:  
  AssetTag:  
  Build:  
  Firmware Version:  
  FruId: v1.0  
  Hardware Version: X04  
  Manufacturer: Ingrasys  
  Name: M2010  
  SerialNumber:  
FW Version: 101.0.3.110  
Name: ocs-am437x
```

```
Network:
  IP Address: 172.16.138.221
  Mac Address: BC:83:85:FA:60:95
Status:
  Health: OK
  LED Status: OFF
  Power: ON
Temperature:
  Humidity: 17.65 RH
  Temperature: 31.99 C
Completion Code: Success
```

## 6.2 Show Manager Health

### Description:

This command gets health status for servers, power supplies, memory usage and temperature etc.

### Syntax:

```
wcscli# show manager health [-s] [-p] [-m] [-h]
```

- b – Show server health
- p – Show PSU health
- m – Show memory Usage
- h – Help, display the correct syntax

### Sample usage:

To get information manager health execute the following command:

```
wcscli# show manager health → Returns all details
```

```
wcscli# show manager health -s → Returns server health details
```

```
wcscli# show manager health -p → Returns power health details
```

```
wcscli# show manager health -m → Returns memory details
```

### Sample output:

```
wcscli# sh manager health -r
```

```
Humidity: 17.65 RH
```

```
Temperature: 32.03 C
```

```
Completion Code: Success
```

```
wcscli# sh manager health -p
```

```
HSC Input-Voltage: 12.0
```



---

HSC Power: 7.05  
HSC Status: Alert{0}  
Completion Code: Success

---

### 6.3 Show Manager Version

**Description:**

This command shows manager version .

**Syntax:**

```
Wcscli# show manager version
```

**Sample output:**

```
DeviceTree: 1.0.7.1  
  
Kernel: 1.0.7.1  
  
PRUFW: 0.9  
  
Package: 1.0.7.1  
  
Rootfs: 1.0.7.1  
  
uboot: 1.0.7.1  
  
Completion Code: Success
```

---

### 6.4 Show Manager Relay

**Description:**

This command gets the power status of the manager (whether ON or OFF) .

**Syntax:**

```
Wcscli# show manager relay
```

**Sample output:**

```
Completion Code : Success  
Relay: ON
```

---

### 6.5 Show Manager LED

**Description:**

This command gets the status of the manager LED (whether ON or OFF) .

---

**Syntax:**

```
Wcscli# show manager led
```

**Sample output:**

```
Completion Code : Success  
Manager LED Status: ON
```

## 6.6 Show Manager Port State

**Description:**

This command gets the status of the port (whether ON or OFF) . The command also returns the status of data save operation (NVDIMM or non-PLP PCIe) using “DatasaveStatus” and also reports the remaining save time in seconds using “DatasaveRemainingTime” when save is pending.

Valid values for DatasaveStatus

- Unknown
- NotApplicable
- SaveNotStarted
- SavePending
- SaveCompleted

DatasaveRemainingTime is reported in seconds when DatasaveStatus is SavePending.

**Syntax:**

```
Wcscli# show manager port state -i <port -id>
```

**Sample output:**

[ Save is not pending ]

```
WcsCli# show manager port state -i 2
```

```
DatasaveStatus: SaveNotStarted  
Port State: On  
Completion Code: Success
```

[ Save operation is pending ]

```
WcsCli# show manager port state -i 2
```

```
DatasaveRemainingTime: 365  
DatasaveStatus: SavePending  
Port State: On  
Completion Code: Success
```

## 6.7 Show Manager Port Presence

### Description:

This command shows port presence state (whether True or False) .

### Syntax:

```
Wcscli# show manager port presence -i <port -id>
```

### Sample output:

```
Completion Code : Success  
Port Presence: True
```

---

## 6.8 Show Manager Port Pairing

### Description:

This command shows power control pairing.

### Syntax:

```
Wcscli# show manager port pairing -i <port -id>
```

### Sample output:

Child Ports:

Parent Ports:

Completion Code: Success

---

## 6.9 Show Manager Port Inventory

### Description:

This command shows port Inventory. Chassis(J2010....) Expander details.

### Syntax:

```
Wcscli# show manager port inventory -i <port -id>
```

### Sample output:

```
WcsCli# show manager port inventory -i 5
```

5:

Expanders Information:

Controller 1:

ID: 00 05 00 05

---

**SAS Expander Addresses:**

1: BF 37 78 02 60 63 01 50

2: BF 36 78 02 60 63 01 50

GUID: 00F01DBC-F540-6083-E741-C05800144C48

MACAddress: F0:1D:BC:F5:40:1F

Completion Code: Success

**6.10 Show Manager Log****Description:**

This command prints rackmanager telemetry log contents.

**Syntax:**

Wcscli# show manager log

**USAGE:**

WcsCli# show manager log -h

```
usage: show manager log [-h] [-b STARTTIME] [-e ENDTIME] [-s STARTID]
                        [-f ENDDID] [-l LOGLEVEL] [-c COMPONENT] [-i
                        DEVICEID]
                        [-p PORTID]
```

**optional arguments:**

```
-h, --help      show this help message and exit
-b STARTTIME    Start time to filter by
-e ENDTIME      End time to filter by
-s STARTID      Start message ID to filter by
-f ENDDID       End message ID to filter by
-l LOGLEVEL     Log level to filter by {0-2}
-c COMPONENT    Component to filter by {0-5}
-i DEVICEID     Device ID to filter by {1-48}
-p PORTID       Port ID to filter by {1-48}
```

**Sample output:**

Child Ports:

Parent Ports:

Completion Code: Success

6.11 Show Manager Inventory

**Description:**  
This command shows manager inventory details.

**Syntax:**  
  
Wcscli# show manager inventory

**Sample output:**

WcsCli# show manager inventory

SW Port	Port State	Present	Slot Id	MAC1	Completion Code
1	ON	True	1	Failure	Failure
2	ON	True	2	BC:83:85:FA:65:8C	Success
3	ON	True	3	BC:83:85:FA:67:A1	Success

6.12 Show Manager scandevice

**Description:**  
This command displays rack manager system information.

**Syntax:**  
  
Wcscli# show manager scandevice

6.13 Show Manager Fru

**Description:**  
  
This command gets the FRU information of the rack and row manager based on the input board name parameter. Default it returns **MB** board FRU data if you don't pass any arguments.

**Syntax:**  
**Rack Manager:**  
Wcscli# show manager fru [-b]{boardname}  
-b – boardnames {mb,pib,acdc}

**Row Manager:**  
  
Wcscli# show manager fru [-b]{boardname}

SW: RM CLI

-b – boardnames {mb,pib}

-h – Help, display the correct syntax

**Sample usage:**

To get the FRU information for the manager, use the following command:

```
wcscli# show manager fru → default returns MB board fru deatails
```

```
wcscli# show manager fru -b pib → returns PIB board fru deatails
```

```
wcscli# show manager fru -b acdc → returns acdc board fru deatails only  
for RACK.
```

**Sample output:**

### 6.14 Show Manager tftp Status

**Description:**

This command gets the status of the TFTP server .

**Syntax:**

```
Wcscli# show manager tftp status
```

**Sample output:**

```
wcscli# sh manager tftp status
```

TFTPSERVICE: Enabled

TFTPSTATUS: Running

Completion Code: Success

### 6.15 Show Manager NFS Status

**Description:**

This command gets the status of the NFS server .

**Syntax:**

```
Wcscli# show manager nfs status
```

**Sample output:**

```
wcscli# sh manager nfs status
```

NFSSERVICE: Enabled

---

NFSStatus: Running  
Completion Code: Success

---

### 6.16 Show PowerMeter Alert

#### Description:

This command shows rack's alert policy.

#### Syntax:

```
Wcscli# show manager powermeter alert [-h]
```

#### Sample output:

```
wcscli# show manager powermeter alert
ChassisPowerAlertLimitInWatts : 19970.0
ChassisPowerDrawnInWatts : 1987.0
Completion Code : Success
IsChassisPowerAlertActive : On
IsChassisPowerAlertEnabled : Off
IsDcThrottleActive : On
IsDcThrottleEnabled : Off
IsPsuPowerAlertLimitActive : 0
IsPsuPowerAlertLimitEnabled : 0
NumPsusAportioningAlert : 0
PsuPowerAlertLimitInWatts : 0
PsuPowerDrawnInWatts : 0
```

---

### 6.17 Show PowerMeter Limit

**Description:** This command shows racks' power limit policy.

#### Syntax:

```
Wcscli# show manager powermeter limit [-h]
```

#### Sample output:

```
ChassisPowerAlertLimitInWatts : 19970.0
Completion Code : Success
```

## 6.18 Show Manager PowerMeter Reading

### Description:

This command shows rack's power readings. Also included are voltage/current readings, power factor and real power draw readings.

### Syntax:

```
wsccli# show manager powermeter reading [-h]
```

### SampleOutput:

```
wsccli# show manager powermeter reading
Feed1Phase1CurrentInAmperes: 9.0
Feed1Phase1PowerInWatts: 2015.0
Feed1Phase1VoltageInVolts: 215.0
Feed1Phase2CurrentInAmperes: 11.0
Feed1Phase2PowerInWatts: 2290.0
Feed1Phase2VoltageInVolts: 217.0
Feed1Phase3CurrentInAmperes: 9.0
Feed1Phase3PowerInWatts: 2015.0
Feed1Phase3VoltageInVolts: 217.0
Feed1PowerStatus: Healthy
Feed2Phase1CurrentInAmperes: 0.0
Feed2Phase1PowerInWatts: 0.0
Feed2Phase1VoltageInVolts: 44.0
Feed2Phase2CurrentInAmperes: 0.0
Feed2Phase2PowerInWatts: 0.0
Feed2Phase2VoltageInVolts: 39.0
Feed2Phase3CurrentInAmperes: 1.0
Feed2Phase3PowerInWatts: 32.0
Feed2Phase3VoltageInVolts: 34.0
Feed2PowerStatus: Faulty (Phase 1 UV Fault, Phase 2 UV Fault, Phase
3 UV Fault)
MaxPowerInWatts: 6923.0
PowerDrawnInWatts: 6352.0
PowerFactor: 0.87
RealPowerInWatts: 5516.0
Completion Code: Success
```

## 6.19 Show Manager PowerMeter Multipolicy Active

### Description:



This command shows multi-policy power configuration for active requestor.

**Syntax:**

```
wsccli# show manager powermeter multipolicy active [-h]
```

**SampleOutput:**

```
wsccli# show manager powermeter multipolicy active
```

```
ActiveRequestorId: 1
AutoRemoveDelay: 30
BladeAlertAction: DoNothing
BladeThrottleLimit: 350
PowercapForceDisable: False
RackDcThrottleEnable: False
RackPowerAlertEnable: True
RackPowerAlertLimitWatts: 7000.0
RemediationAction: DoNothing
RequestorId: 1
RequestorIdActive: True
ThrottleDuration: 20
TotalConfiguredRequestors: 1
Completion Code: Success
```

## 6.20 Show Manager PowerMeter Multipolicy Single

**Description:**

This command shows multi-policy power configuration for the specified requestor.

**Syntax:**

```
wsccli# show manager powermeter multipolicy single -i requestor-id [-h]
```

**SampleOutput:**

```
wsccli# show manager powermeter multipolicy single -i 1
```

```
ActiveRequestorId: 1
AutoRemoveDelay: 30
BladeAlertAction: DoNothing
BladeThrottleLimit: 350
PowercapForceDisable: False
RackDcThrottleEnable: False
RackPowerAlertEnable: True
```

```
RackPowerAlertLimitWatts: 7000.0
RemediationAction: DoNothing
RequestorId: 1
RequestorIdActive: True
ThrottleDuration: 20
TotalConfiguredRequestors: 1
Completion Code: Success
```

## 6.21 Show Manager PowerMeter Multipolicy All

### Description:

This command shows multi-policy power configuration for all configured requestors.

### Syntax:

```
wsccli# show manager powermeter multipolicy all [-h]
```

### SampleOutput:

```
wsccli# show manager powermeter multipolicy all
requestors:
  1:
    AutoRemoveDelay: 30
    BladeAlertAction: DoNothing
    BladeThrottleLimit: 350
    PowercapForceDisable: False
    RackDcThrottleEnable: False
    RackPowerAlertEnable: True
    RackPowerAlertLimitWatts: 7000.0
    RemediationAction: DoNothing
    RequestorId: 1
    RequestorIdActive: True
    ThrottleDuration: 20
    TotalConfiguredRequestors: 2
  2:
    AutoRemoveDelay: 10
    BladeAlertAction: FastThrottle
    BladeThrottleLimit: 300
    PowercapForceDisable: False
    RackDcThrottleEnable: False
    RackPowerAlertEnable: False
    RackPowerAlertLimitWatts: 5600.0
    RemediationAction: RearmOnly
```

```
RequestorId: 2
RequestorIdActive: False
ThrottleDuration: 30
TotalConfiguredRequestors: 2
Completion Code: Success
```

---

## 6.22 Show Manager Power Meter Version

### Description:

This command shows the manager's PRU version.

### Syntax:

```
wcscli# show manager powermeter version [-h]
```

### Sample Usage:

```
wcscli# show manager power pruversion
```

### Sample OutPut:

```
Completion Code : Success
PRUFirmwareVersion : 0.5
```

---

## 6.23 Show Manager Power Reading

### Description:

This command shows the manager's power reading.

### Syntax:

```
wcscli# show manager power reading [-h]
```

### Sample Usage:

```
wcscli# show manager power reading
```

### Sample OutPut:

```
PowerInWatts: 6.0
Completion Code: Success
```

---

## 6.24 Show Manager Power Status

### Description:

This command shows the manager's power status.

---

**Syntax:**

```
wcscli# show manager power status [-h]
```

**Sample Usage:**

```
wcscli# show manager power status
```

**Sample OutPut:**

```
wcscli# sh manager power status
```

```
HSC Status: Healthy
```

```
PSU A Status: Faulty
```

```
PSU B Status: Faulty
```

```
Completion Code: Success
```

---

## 6.25 Show Manager NTP Status

**Description:**

Show the status of the NTP service.

**Syntax:**

```
wcscli# show manager ntp status [-h]
```

**Sample Usage:**

```
wcscli# show manager ntp status
```

**Sample OutPut:**

```
wcscli# sh manager ntp status
```

```
NTPService: Enabled
```

```
NTPState: Running
```

```
Completion Code: Success
```

---

## 6.26 Show Manager NTP Server

**Description:**

Show the time server being used by the NTP service

**Syntax:**

```
wcscli# show manager ntp server [-h]
```

---

**Sample Usage:**

```
wcscli# show manager ntp status
```

**Sample OutPut:**

```
wcscli# sh manager ntp server
```

```
NTPServer: 127.127.1.0
```

```
Completion Code: Success
```

---

## 6.27 Show Manager ITP Status

**Description:**

This command shows remote ITP service info.

**Syntax:**

```
wcscli# show manager itp status [-h]
```

**Sample Usage:**

```
wcscli# show manager itp st
```

**Sample OutPut:**

```
WcsCli# show manager itp status
```

```
ITPService: Enabled
```

```
ITPState: Running
```

```
Completion Code: Success
```

---

## 6.28 Show manager throttle local

**Description:**

This command shows rack manager throttle control status

**Syntax:**

```
wcscli# show manager throttle [-h] local
```

**Sample Usage:**

```
wcscli# show manager throttle local
```

**Sample OutPut:**

Local Bypass: False

Local Enable: True

Completion Code: Success

---

## 6.29 Show manager throttle count

### Description:

This command shows rack manager throttle count

### Syntax:

```
wcscli# show manager throttle [-h] count
```

The 'remote' option is only available for the row manager

### Sample Usage:

```
wcscli# show manager throttle count
```

### Sample OutPut:

ThrottleCount: 0

ThrottleCountDuration: 4:23:25.463960

Completion Code: Success

---

## 6.30 Show manager throttle row

### Description:

This command shows row manager throttle control status

### Syntax:sh

```
wcscli# show manager throttle [-h] {local,count, row}
```

The "row" option is only available for the row manager

### Sample Usage:

```
wcscli# show manager throttle row
```

### Sample OutPut:

Row Bypass: False

Row Enable: True

---

Completion Code: Success

---

### 6.31 Show manager recovery (boot strap)

**NOTE:** This command is only available for the ROW Manager

**Description:**

This command shows row manager boot strap status

**Syntax:**

```
wcscli# show manager recovery <-i port-id>[-h]
```

**Sample Usage:**

```
wcscli# show manager recovery -i 1
```

**Sample OutPut:**

Boot Strap: Normal

Completion Code: Success

---

### 6.32 Show Manager Session list

**Description:**

This command shows active manager ssh sessions

**Syntax:**

```
wcscli# show manager session list [-h]
```

**Sample usage:**

```
wcscli# sh manager session list
```

**Sample output:**

sessions:

1:

ClientIP: 192.168.150.151

ClientPort: 57360

ID: 389

2:

ClientIP: 192.168.150.163

---

ClientPort: 41810

ID: 8991

3:

ClientIP: 192.168.150.43

ClientPort: 59181

ID: 8984

Completion Code: Success

---

### 6.33 Show Manager Tftp list

#### Description:

This command list files under TFTP location

#### Syntax:

```
wsccli# show manager tftp list [-h]
```

#### Sample usage:

```
wsccli# show manager tftp list
```

#### Sample output:

TFTPFiles:

1:

Name: AuditLog

2:

Name: DebugLog

3:

Name: TelemetryLog

4:

Name: audit.log

5:

Name: auditlog

6:



```
Name: hello
7:
Name: tftpfile2
8:
Name: winpe_amd64.iso
9:
Name: wwwram
```

```
Completion Code: Success
```

---

### 6.34 Show Manager Time

**Description:**

This command displays manager system time and date in UTC

**Syntax:**

```
wcscli# show manager time [-h]
```

**Sample usage:**

```
wcscli# show manager time
```

**Sample output:**

```
DateTime: 2017-02-21 19:29:59
```

```
Completion Code: Success
```

---

### 6.35 Show Manager Type

**Description:**

This command displays manager type

**Syntax:**

```
wcscli# show manager type [-h]
```

**Sample usage:**

```
wcscli# show manager type
```

**Sample output:**

---

Manager Type: PMDU\_RACKMANAGER

Completion Code: Success

---

### 6.36 Show Manager menu

**Description:**

This command tells if the recovery menu that is displayed at Rack Manager U-Boot bootup is on or off

**Syntax:**

```
wcsccli# show manager recoverymenu [-h]
```

**Sample usage:**

```
wcsccli# show manager menu
```

**Sample output:**

Recovery Menu: Enabled

Completion Code: Success

---

### 6.37 Show Manager fwupdate status

**Description:**

This command shows most recent firmware update status

**Syntax:**

```
wcsccli# show manager fwupdate status [-h]
```

**Sample usage:**

```
wcsccli# show manager fwupdate status
```

**Sample output:**

FWUpdate status: Complete

Completion Code: Success

---

### 6.38 Show Manager fwupdate list

**Description:**

This command lists versions of available Rack Manager firmware packages that could be used for recovery

---

**Syntax:**

```
wcscli# show manager fwupdate list [-h]
```

**Sample usage:**

```
wcscli# show manager fwupdate list
```

**Sample output:**

```
active: 1.0.5.458
```

```
factory: 1.0.5.452
```

```
rollback: 1.0.5.458
```

```
upgrade: N/A
```

```
Completion Code: Success
```

---

### 6.39 Show Manager jtag status

**Description:**

This command shows JTAG service status

**Syntax:**

```
wcscli# show manager jtag status [-h]
```

**Sample usage:**

```
wcscli# show manager jtag status
```

**Sample output:**

```
wcscli# show manager jtag status
```

```
JTAGService: Enabled
```

```
JTAGState: Stopped
```

```
Completion Code: Success
```

---

### 6.40 Show manager powermeter calibration

**Description:**

This command shows the calibration values set in the PRU (powermeter)

**Syntax:**

```
WcsCli# show manager powermeter calibration
```

**Sample Usage:**

```
WcsCli# show manager powermeter calibration
```

```
Current:
```

```
Feed1:
```

```
Phase1:
```

```
Gain: 17123
```

```
Offset: 33143
```

```
Phase2:
```

```
Gain: 17101
```

```
Offset: 33117
```

```
Phase3:
```

```
Gain: 17190
```

```
Offset: 33174
```

```
Feed2:
```

```
...
```

```
Voltage:
```

```
Feed1:
```

```
Phase1:
```

```
Gain: 17683
```

```
Offset: 33165
```

```
Phase2:
```

```
Gain: 17912
```

```
...
```

#### 6.41 Show manager throttle count

**Description:**

This command displays the number of throttle events since the last reset time. This command also shows the last reset time.

**Syntax:**

```
WcsCli# show manager throttle count
```

**Sample Usage:**

```
WcsCli# show manager throttle count
```

```
ThrottleCount: 12
```

```
ThrottleCountDuration: 6 days 1:25:42.628122
```

```
Completion Code: Success
```

#### 6.42 Show manager pubsub config

**Description:**

This command displays pubsub settings “Brokers”, “Topic”, and “Ssl”.

**Syntax:**

```
WcsCli# show manager pubsub
```

**Sample Usage:**

```
WcsCli# show manager pubsub
Brokers: 10.10.10.500
Ssl: 0
Topic: CSIRM
Completion Code: Success
```

### 6.43 Show manager userconfigs

**Description:**

This command displays user configuration parameter settings.

**Syntax:**

```
WcsCli# show manager userconfigs
```

**Sample Usage:**

```
WcsCli# show manager userconfigs
userconfigs:
  1:
    PollCriticalDataInSeconds
    60
  2:
    PollNonCriticalDataInMinutes
    5
  3:
    PollNonCriticalLogInMinutes
    30
  4:
    RelayCableOverride
    0
  5:
    MsrCpuRetryReadLogEnable
    0
  6:
    ConfigSerialPortFlowControl
    0
  7:
    PowercapMultiPolicyEnable
    1
  8:
    PollPowerInSeconds
    600
  9:
```

```

    HpSdDcPowerOpConfig
    0
10:
    RustEnableConfig
    0
11:
    sshAdvancedCipherEnable
    0
12:
    RealPowerThrottleEnable
    0
13:
    MsrDumpEnable
    0
14:
    AcdDumpEnable
    1
Completion Code: Success

```

#### 6.44 **how manager serialstats**

##### **Description:**

This command will run serial terminal statistics based on user requested timed interval.

##### **Syntax:**

WcsCli# show manager serialstates -i interval( 1 - 60 secs)

##### **Sample Usage:**

WcsCli# show manager serialstats -i 5

```

cts: 0 dsr: 0 rng: 0 dcd: 0 rx: 0 tx: 0 frame error 0 overruns 0 parity: 0 break: 0 buffer overrun: 0
cts: 0 dsr: 0 rng: 0 dcd: 0 rx: 0 tx: 0 frame error 0 overruns 0 parity: 0 break: 0 buffer overrun: 0
cts: 0 dsr: 0 rng: 0 dcd: 0 rx: 0 tx: 0 frame error 0 overruns 0 parity: 0 break: 0 buffer overrun: 0
cts: 0 dsr: 0 rng: 0 dcd: 0 rx: 0 tx: 0 frame error 0 overruns 0 parity: 0 break: 0 buffer overrun: 0
cts: 0 dsr: 0 rng: 0 dcd: 0 rx: 0 tx: 0 frame error 0 overruns 0 parity: 0 break: 0 buffer overrun: 0

```

Completion Code: Success

#### 6.45 **Show manager serialcheck**

##### **Description:**

This command will run the serial terminal health based on user requested count.

##### **Syntax:**

WcsCli# show manager serialcheck -l loop < 1 – 1000 times>

**Sample Usage:**

```
WcsCli# show manager serialcheck -l 10
1+0 records in
1+0 records out
Needed 0 reads 1 writes loops 27 / 100
```

Completion Code: Success

#### 6.46 **Show manager ssh status**

**Description:**

This command will display the SSH cipher security configuration level along with health, process status and user config details

**Syntax:**

```
WcsCli# show manager ssh status
```

**Sample Usage:**

```
WcsCli# show manager ssh status
Ciphers: aes256-cbc,aes192-cbc,aes128-cbc,3des-cbc
KexAlgorithms: diffie-hellman-group1-sha1,ecdh-sha2-nistp521,ecdh-sha2-nistp384,ecdh-sha2-
nistp256
MACs: hmac-sha1,umac-64@openssh.com,hmac-ripemd160
SSHCompliance: OK
SSHProcess: Running
SSHUserconfig: Disabled
Completion Code: Success
```

---

### *Set Commands*

- [set manager led on](#)
- [set manager led off](#)
- [set manager relay on](#)
- [set manager relay off](#)
- [set manager port on](#)
- [set manager port off](#)
- [set manager port pairing add](#)
- [set manager port pairing remove](#)
- [set manager log clear](#)
- [set manager fru](#)
- [set manager power clear faults](#)
- [set manager powermeter Alert](#)
- [set manager powermeter limit](#)
- [set manager powermeter multipolicy](#)
- [set manager powermeter clearmax](#)
- [set manager powermeter clearfaults](#)
- [set manager tftp start](#)
- [set manager tftp stop](#)
- [set manager nfs start](#)
- [set manager nfs stop](#)
- [Set manager ntp start](#)
- [set manager ntp stop](#)
- [set manager ntp restart](#)
- [set manager ntp enable](#)
- [set manager ntp disable](#)
- [set manager ntp server](#)
- [Set manager itp start](#)



- [set manager itp stop](#)
- [set manager itp restart](#)
- [set manager itp enable](#)
- [set manager itp disable](#)
- [set manager hostname](#)
- [set manager throttle local](#)
- [set manager throttle count clear](#)
- [set manager throttle force enable](#)
- [set manager throttle row](#) (ROW manager command)
- [set manager recovery on](#)
- [set manager recovery off](#)
- [set manager session kill](#)
- [set manager tftp get](#)
- [set manager tftp put](#)
- [set manager tftp delete](#)
- [set manager time](#)
- [set manager menu on](#)
- [set manager menu off](#)
- [set manager fwupdate](#)
- [set manager defaults](#)
- [Set manager jtag start](#)
- [set manager jtag stop](#)
- [set manager jtag restart](#)
- [set manager jtag enable](#)
- [set manager jtag disable](#)
- [Set manager powermeter calibration](#)
- [Set manager throttle force enable](#)
- [Set manager throttle count clear](#)
- [Set manager pubsub](#)
- [Set manager userconfigs](#)
- [Set manager ssh config](#)

### *User Commands*

- [show user info](#)
- [set user add](#)
- [set user update](#)
- [set user delete](#)

---

#### 6.47 Set Manager LED On

**Description:**

This command turns the Manger LED ON.

**Syntax:**

```
Wcscli# set manager led on
```

**Sample output:**

Completion Code : Success

---

#### 6.48 Set Manager LED Off

**Description:**

This command turns the Manger LED OFF.

**Syntax:**

```
Wcscli# set manager led off
```

**Sample output:**

Completion Code : Success

---

#### 6.49 Set Manager Relay On

**Description:**

This command turns the manager AC sockets (TOR switches) ON.

**Syntax:**

```
Wcscli# set manager relay on <-p port_no> [-h]
```

-p – Port number of interest

-h – Help, display the correct syntax

**Sample usage:**

To turn ON the manager AC sockets for port 1, use the following command:

```
wcscli# set manager relay on -p 1
```

**Sample output:**

Completion Code : Success

---

#### 6.50 Set Manager Relay Off

---

**Description:**

This command turns the manager AC sockets (TOR switches) OFF.

**Syntax:**

```
Wcscli# set manager relay off <-p port_no> [-h]
```

-p – Port number of interest

-h – Help, display the correct syntax

**Sample usage:**

To turn OFF the manager AC sockets for port 1, use the following command:

```
wcscli# set manager relay off -p 1
```

**Sample output:**

Completion Code : Success

---

## 6.51 Set Manager Port On

**Description:**

This command turns the AC outlet power ON for the servers if you run from Rack Manager.

This command turns the AC outlet power ON for the Racks if you run from Row Manager.

Refer to the sample output section for behavior of this command when save is pending for a blade in a PMDU rack manager.

**Syntax:**

```
wcscli# set manager port on <-i port_index> [-h]
```

-i – port index (1-48)

-h – Help, display the correct syntax

**Sample usage:**

To turn the power ON on port 3, use the following command:

```
wcscli# set manager port on -i 3
```

**Sample output:**

Completion Code : Success

When save is pending the command with return failure along with amount of time to wait for save completion. The command should be re-attempted after this wait time.

```
WcsCli# set manager port on -i 2
```

Status Description: Operation not permitted for blade (2) due to pending save, retry after (365) seconds

Completion Code: Failure

User may use the “show manager port state” command to get save status.

---

---

## 6.52 Set Manager Port Off

**Description:**

This command turns the AC outlet power OFF for the servers if you run from Rack Manager.

This command turns the AC outlet power OFF for the Racks if you run from Row Manager.

Refer to the sample output section for behavior of this command when save is pending for a blade in a PMDU rackmanager.

**Note:**

This command by default will attempt to perform a datasafe power off operation – perform NVDIMM save and then do power off when it is complete. The command completion might take as much as 6 minutes. The status of the save operation can be obtained via the show manager port state command.

User must use the force option to forcefully power off the port without performing any datasafe operation – THIS MIGHT RESULT IN DATA LOSS.

**Syntax:**

```
wsccli# set manager port off <-i port_index> [-h] [-f]
```

-i – port index (1-48)

-h – Help, display the correct syntax

-f, --force force port off

**Sample usage:**

To forcefully turn the power OFF without performing datasafe operation, on port 3, use the following command:

```
wsccli# set manager port off -i 3 -f
```

**Sample output:**

Completion Code : Success

Note: This command may take as much as 5 seconds to 6 minutes to complete depending on the amount of software NVDIMM configured in the system.

```
Wsccli# set manager port off -i 3
```

Sample output:

Completion Code: Success

When save is pending,

---

WcsCli# set manager port off -i 2

Status Description: Operation not permitted for blade (2) due to pending save, retry after (365) seconds

Completion Code: Failure

---

### 6.53 Set Manager Port pairing add

#### Description:

This command adds a power control pairing. After a port has been paired with a parent port, it will be turned on/off when the parent port is turned on/off.

Some devices allow finer control over what can be turned on/off in addition to the main power for the whole system. For these devices, function pairing can specify which internal function should be paired with the parent port. The valid IDs and ID mapping of functions are device dependent.

#### Syntax:

```
usage: set manager port pairing add [-h] -i PORT_ID -c CHILD_PORT
      [-f CHILD_FUNC]
```

#### optional arguments:

```
-h, --help      show this help message and exit
-f CHILD_FUNC   Child device function ID
```

#### required arguments:

```
-i PORT_ID      Parent port ID
-c CHILD_PORT   Child port ID
```

#### Sample usage:

```
wcscli# set manager port pairing add -i 3 -c 2
```

#### Sample output:

Completion Code : Success

#### Notes:

- A parent can be paired to any number of children. These children can also be paired to any number of children. This excludes functions. Functions cannot be paired to children.
  - A port can only be paired to a single parent unless function pairing is being used. In that case, each function can have a different parent.
  - No loops are allowed in the hierarchy. Meaning port A cannot be a child of port B if there exists a pairing hierarchy that makes B a descendant of A.
  - A port cannot be paired without specifying a function if there already exists a pairing to a function
-

on that port.

- A port function cannot be paired if there already exists a pairing to the port without a function.
- Power off commands start at the root of the hierarchy and work down. Power on goes in reverse.
- Power on/off commands will stop at the first failure, which may leave some of the hierarchy in the new state and some in the old state when there is an error.
- Function pairing only works for devices that support function pairing. If the server type in a slot cannot be determined at the time of pairing, the function pairing is allowed to succeed. If a device is later added to that slot that doesn't support function pairing, power control operations on the parent of that misconfigured port will fail.

---

#### 6.54 Set Manager Port pairing remove

##### Description:

This command removes a power control pairing.

##### Syntax:

```
usage: set manager port pairing remove [-h] -i PORT_ID -c CHILD_PORT
                                         [-f CHILD_FUNC]
```

optional arguments:

```
-h, --help      show this help message and exit
-f CHILD_FUNC   Child device function ID
```

required arguments:

```
-i PORT_ID      Parent port ID
-c CHILD_PORT   Child port ID
```

##### Sample usage:

```
wsccli# set manager port pairing remove -i 3 -c 2
```

##### Sample output:

Completion Code : Success

---

#### 6.55 Set Manager Log Clear

##### Description:

This command clears all log entries.

##### Syntax:

```
wsccli# set manager log clear [-h]
```

---

-h – Help, display the correct syntax

**Sample usage:**

```
wsccli# set manager log clear
```

**Sample output:**

Completion Code : Success

---

## 6.56 Set Manager Fru

**Description:**

This command is used to update the Mnager FRU information

**Syntax:**

```
wsccli# set manager fru [-h]
```

-h – Help, display the correct syntax

**Sample usage:**

```
wsccli#
```

```
usage: set manager fru [-h] -b {mb,pib,acdc} -f FILENAME
```

optional arguments:

-h, --help show this help message and exit

required arguments:

-b {mb,pib,acdc} Manager board name

-f FILENAME fru file path

**Sample output:**

Completion Code : Success

---

## 6.57 Set Manager ClearFaults

**Description:**

This command clears manager power faults.

**Syntax:**

```
Wsccli# set manager power faults [-h]
```

-h – Help, display the correct syntax

**Sample usage:**

```
wsccli# set manager power faults
```

**Sample output:**

---

---

Completion Code : Success

---

## 6.58 Set Manager PowerMeter Alert

### Description:

This command sets rack power alert policy.

### Syntax:

```
Wcscli# set manager powermeter alert [-h] -e poweralertpolicy
                                         -d dcthrottlealertpolicy
                                         -p powerlimit
```

-e poweralertpolicy poweralertpolicy (0 disable, 1 enable)  
-d dcthrottlealertpolicy dcthrottlealertpolicy (0 disable, 1 enable)  
-p powerlimit power limit in watts  
-h Help display the correct syntax

### Sample usage:

```
wcscli# set manager powermeter alert -e 1 -d 0 -p 500
```

### Sample output:

Completion Code : Success

---

## 6.59 Set Manager PowerMeter Limit

### Description:

This command sets rack power limit.

### Syntax:

```
Wcscli# set manager powermeter limit [-h] -p power limit
-p powerlimit    power limit in watts
-h Help          display the correct syntax
```

### Sample usage:

```
wcscli# set manager powermeter limit -p 400
```

### Sample output:

Completion Code : Success

---

## 6.60 Set Manager PowerMeter Multipolicy



**Description:**

This command sets multi-policy power configuration for rack and systems .

**Syntax:**

```
Wcscli# set manager powermeter multipolicy [-h] -i requestor-id
-a rack-alert-policy
-t dc-throttle-enable
-l rack-power-limit
-p blade-power-limit
-e alert-action -f
throttle-duration
-d remove-delay
[-r remediation-action]

-h, --help          show this help message and exit
-i requestor-id      Requestor ID (0-32)
-a rack-alert-policy Rack level alert policy (-1: forceDisable, 0:
Disable,1: Enable)
-t dc-throttle-enable Rack level DC throttle (0: disable, 1: Enable)
-l rack-power-limit  Rack level power limit
-p blade-power-limit power limit in watts
-e alert-action       alert action, 0:nothing, 1:power limit only,
2:throttle and limit
-f throttle-duration fast throttle duration in milliseconds
-d remove-delay       auto remove power limit delay in seconds
-r remediation-action remediation action, 0:nothing, 1:remove limit and
re-arm alert, 2:re-arm alert
```

**Sample usage:**

```
wcscli# set manager powermeter multipolicy -i 1 -a 1 -t 0 -l 6000 -p 300
-e 1 -f 10 -d 20 -r 1
```

**Sample output:**

Completion Code : Success

**6.61 Set Manager PowerMeter ClearMax**

**Description:**

This command clears rack max power.

**Syntax:**

```
Wcscli# set manager powermeter max [-h]
```

-h – Help, display the correct syntax

**Sample usage:**

```
wcscli# set manager powermeter max
```

SW: RM CLI

**Sample output:**

Completion Code : Success

---

## 6.62 Set Manager PowerMeter ClearFaults

**Description:**

This command clears rack power faults.

**Syntax:**

```
Wcscli# set manager powermeter faults [-h]
```

-h – Help, display the correct syntax

**Sample usage:**

```
wcscli# set manager powermeter faults
```

**Sample output:**

Completion Code : Success

---

## 6.63 Set Manager TFTP Start

**Description:**

This command starts the TFTP server on the rack manager.

**Syntax:**

```
Wcscli# set manager tftp start[-h]
```

-h Help                      display the correct syntax

**Sample usage:**

```
wcscli# set manager tftp start
```

**Sample output:**

Completion Code : Success

---

## 6.64 Set Manager TFTP Stop

**Description:**

This command stops the TFTP server on the rack manager.

---

**Syntax:**

```
Wcscli# set manager tftp stop[-h]
```

-h Help                      display the correct syntax

**Sample usage:**

```
wcscli# set manager tftp stop
```

**Sample output:**

Completion Code : Success

---

### 6.65 Set Manager NFS Start

**Description:**

This command starts the NFS server on the rack manager.

**Syntax:**

```
Wcscli# set manager nfs start[-h]
```

-h Help                      display the correct syntax

**Sample usage:**

```
wcscli# set manager nfs start
```

**Sample output:**

Completion Code : Success

---

### 6.66 Set Manager NFS Stop

**Description:**

This command stops the NFS server on the rack manager.

**Syntax:**

```
Wcscli# set manager nfs start[-h]
```

-h Help                      display the correct syntax

**Sample usage:**

```
wcscli# set manager nfs start
```

**Sample output:**

Completion Code : Success

---

---

### 6.67 Set Manager NTP Start

**Description:**

Start the NTP service.

**Syntax:**

```
wcscli# set manager ntp start [-h]
```

-h – Help, display the correct syntax

**Sample usage:**

```
wcscli# set manager ntp start
```

**Sample output:**

---

### 6.68 Set Manager NTP Stop

**Description:**

Stop the NTP service.

**Syntax:**

```
wcscli# set manager ntp stop [-h]
```

-h – Help, display the correct syntax

**Sample usage:**

```
wcscli# set manager ntp stop
```

**Sample output:**

---

### 6.69 Set Manager NTP Restart

**Description:**

Restart the NTP service

**Syntax:**

```
wcscli# set manager ntp restart [-h]
```

-h – Help, display the correct syntax

**Sample usage:**

```
wcscli# set manager ntp restart
```

---

---

**Sample output:**

---

### 6.70 Set Manager NTP Enable

**Description:**

Configure the NTP service to start automatically on boot

**Syntax:**

```
wsccli# set manager ntp enable [-h]
```

-h – Help, display the correct syntax

**Sample usage:**

```
wsccli# set manager ntp enable
```

**Sample output:**

---

### 6.71 Set Manager NTP Disable

**Description:**

Configure the NTP service to not start automatically on boot

**Syntax:**

```
wsccli# set manager ntp disable [-h]
```

-h – Help, display the correct syntax

**Sample usage:**

```
wsccli# set manager ntp disable
```

**Sample output:**

---

### 6.72 Set Manager NTP Server

**Description:**

Set the time server to be used by the NTP service

**Syntax:**

```
wsccli# set manager ntp server [-h] -s <server IP>
```

---

---

-s -> The IP address of the time server.

-h – Help, display the correct syntax

**Sample usage:**

```
wsccli# set manager ntp server -s <server-ip>
```

**Sample output:**

---

### 6.73 Set Manager ITP Start

**Description:**

This command starts the ITP service.

**Syntax:**

```
Wsccli# set manager itp start [-h]
```

-h – Help, display the correct syntax

**Sample usage:**

```
wsccli# set manager itp start
```

**Sample output:**

Completion Code: Success

---

### 6.74 Set Manager ITP Stop

**Description:**

This command stops the ITP service.

**Syntax:**

```
Wsccli# set manager itp stop [-h]
```

-h – Help, display the correct syntax

**Sample usage:**

```
wsccli# set manager itp stop
```

**Sample output:**

Completion Code: Success

---

### 6.75 Set Manager ITP Restart

**Description:**

This command restarts the ITP service.

**Syntax:**

```
wsccli# set manager itp restart [-h]
```

-h – Help, display the correct syntax

**Sample usage:**

```
wsccli# set manager itp restart
```

**Sample output:**

Completion Code: Success

---

### 6.76 Set Manager ITP Enable

**Description:**

This command will enable the ITP service.

**Syntax:**

```
wsccli# set manager ntp enable [-h]
```

-h – Help, display the correct syntax

**Sample usage:**

```
wsccli# set manager ntp enable
```

**Sample output:**

Completion Code: Success

---

### 6.77 Set mSet Manager ITP Disable

**Description:**

This command will disable the ITP service.

**Syntax:**

```
wsccli# set manager itp disable [-h]
```

-h – Help, display the correct syntax

**Sample usage:**

```
wcscli# set manager itp disable
```

**Sample output:**

Completion Code: Success

---

### 6.78 Set Manager jtag Start

**Description:**

This command starts the JTAG service.

**Syntax:**

```
wcscli# set manager jtag start [-h]
```

-h – Help, display the correct syntax

**Sample usage:**

```
wcscli# set manager jtag start
```

**Sample output:**

Completion Code: Success

---

### 6.79 Set Manager jtag Stop

**Description:**

This command stops the jtag service.

**Syntax:**

```
wcscli# set manager jtag stop [-h]
```

-h – Help, display the correct syntax

**Sample usage:**

```
wcscli# set manager jtag stop
```

**Sample output:**

Completion Code: Success

---

### 6.80 Set Manager jtag Restart

**Description:**



---

This command restarts the jtag service.

**Syntax:**

```
wcscli# set manager jtag restart [-h]
```

-h – Help, display the correct syntax

**Sample usage:**

```
wcscli# set manager jtag restart
```

**Sample output:**

Completion Code: Success

---

### 6.81 Set Manager jtag Enable

**Description:**

This command will enable the jtag service.

**Syntax:**

```
wcscli# set manager jtag enable [-h]
```

-h – Help, display the correct syntax

**Sample usage:**

```
wcscli# set manager jtag enable
```

**Sample output:**

Completion Code: Success

---

### 6.82 Set Manager jtag Disable

**Description:**

This command will disable the jtag service.

**Syntax:**

```
wcscli# set manager jtag disable [-h]
```

-h – Help, display the correct syntax

**Sample usage:**

```
wcscli# set manager jtag disable
```

---

**Sample output:**

Completion Code: Success

---

### 6.83 Set Manager Hostname

**Description:**

Set the hostname of the system

**Syntax:**

```
wcscli# set manager hostname [-h] -n <hostname>
```

-n -> The name to use as the system hostname.

-h – Help, display the correct syntax

**Sample usage:**

```
wcscli# set manager hostname -n ocsmanager01
```

**Sample output:**

---

### 6.84 Set Manager throttle local

**Description:**

This command controls rack manager throttle configuration.

**Syntax:**

```
wcscli# set manager throttle local {bypass, enable} -e {0,1}
```

1. Set the state of the 4 throttle GPIOs. The -e argument indicates the disable/enable state of the GPIO.
2. Local and remote GPIOs each have a 'bypass' and 'enable' option.

**usage:** set manager throttle local bypass [-h] -e {0,1}

optional arguments:

-h, --help show this help message and exit

required arguments:

-e {0,1} Disable (0)/Enable (1)

**Sample output:**

Completion Code: Success

---

### 6.85 Set Manager throttle count

**Description:**

This command controls rack manager throttle count.

This command clears the throttle count.

**Syntax:**

**set manager throttle count [-h] {clear} ...**

positional arguments:

{clear} throttle count clear action  
clear This command clears throttle count

optional arguments:

-h, --help show this help message and exit

**Usage:**

**Wcscli# set manager throttle count clear**

**Sample output:**

Completion Code: Success

**6.86 Set Manager throttle force****Description:**

This command forces rack manager throttle.

**Syntax:**

**set manager throttle force [-h] {enable} -e {0,1}...**

usage: set manager throttle force enable [-h] -e {0,1}

optional arguments:

-h, --help show this help message and exit

required arguments:

-e {0,1} Disable (0)/Enable (1)

**Usage:**

**Wcscli# set manager throttle force enable -e 0**

**Sample output:**

Completion Code: Success

**6.87 Set Manager throttle row**

**The 'row' option is only available for the row manager**

**Description:**

This command controls row manager throttle configuration.

**Syntax:**

```
wsccli# set manager throttle row {bypass, enable} -e {0,1}
```

usage: set manager throttle row bypass [-h] -e {0,1}

optional arguments:

-h, --help show this help message and exit

required arguments:

-e {0,1} Disable (0)/Enable (1)

-h – Help, display the correct syntax

**Sample usage:**

```
wsccli# set manager throttle row bypass -e 0
```

```
wsccli# set manager throttle row local -e 0
```

**ROW MANAGER Throttle Commands:**

```
wsccli# set manager throttle row bypass -e 0
```

```
wsccli# set manager throttle local bypass -e 0
```

```
wsccli# set manager throttle local enable -e 0
```

```
wsccli# set manager throttle row enable -e 0
```

```
wsccli# set manager throttle count clear
```

```
wsccli# set manager throttle force
```

**Sample output:**

Completion Code: Success

---

## 6.88 Set Manager Recovery On

**NOTE:** This command is only available for the ROW Manager

**“ON :** This option forces the rack manager into network recovery boot.

**“OFF :** This option sets the rack manager to normal boot.”

**Note:** A power reset is required for the command to take effect.”

---

**[STEPS]**

1. Set Network boot ON
2. Power Cycle target Rack Manager
3. Wait for Imaging to start
4. Set Network boot OFF

If you don't set it off it will perpetually boot to network.

**Description:**

This command sets bootstrap on for Rack Manager.

**Syntax:**

```
Wcscli# set manager recovery on [-h] -i <port-id>
```

-i -> Port-Id Manager power port id <1 to 24>.

-h – Help, display the correct syntax

**Sample usage:**

```
wcscli# set manager recovery on -i 1
```

**Sample output:**

Completion Code : Success

---

## 6.89 Set Manager Recovery Off

**NOTE: This command is only available for the ROW Manager**

**Description:**

This command sets bootstrap off for Rack Manager.

**Syntax:**

```
Wcscli# set manager recovery off [-h] -i <port-id>
```

-i -> Port-Id Manager power port id <1 to 24>.

-h – Help, display the correct syntax

**Sample usage:**

```
wcscli# set manager recovery off -i 1
```

**Sample output:**

Completion Code : Success

---

## 6.90 Set Manager Session Kill

**Description:**

---

Kills an existing active SSH session in the Rack manager

Note:

- Use show manager session list command to obtain the session id

**Syntax:**

```
wsccli# set manager session kill [-h] -s <session-id>
```

**Sample usage:**

```
wsccli# set manager session kill -s 15405
```

**Sample output:**

Completion Code : Success

---

## 6.91 Set Manager Tftp Delete

**Description:**

Deletes the specified file from the TFTP location in the RM

Note:

- Use show manager tftp list command to obtain the filename

**Syntax:**

```
wsccli# set manager tftp delete [-h] -f <filename>
```

**Sample usage:**

```
wsccli# set manager tftp delete -f tftpfile1
```

**Sample output:**

Completion Code : Success

---

## 6.92 Set Manager Tftp/Sftp Get

**Description:**

Allows to download the specified file to the RM from a specified TFTP or SFTP server

Note:

- Use show manager tftp list to see the downloaded file (after execution of this command)
-

**Syntax:**

```
Wsccli# set manager tftp get [-h] -f <filename in the tftp server> -s <server-ip>
```

```
Wsccli# set manager sftp get [-h] -f <filename in the sftp server> -s <server-ip> -u <user>
```

**Sample usage:**

```
wsccli# set manager tftp get -s 172.16.138.219 -f myfile
```

```
wsccli# set manager sftp get -s 172.16.138.219 -f myfile -u admin
```

**Sample output:**

Completion Code : Success

---

### 6.93 Set Manager Tftp/Sftp Put

**Description:**

Allows to upload the specified target file from the RM to the specified TFTP or SFTP server location

**Syntax:**

```
Wsccli# set manager tftp put [-h] -f <filename in the tftp server> -s <server-ip> -t <target>
```

```
Wsccli# set manager sftp put [-h] -f <filename in the sftp server> -s <server-ip> -t <target> -u <user>
```

Available targets:

- auditlog
- debuglog
- telemetrylog
- restlog
- firmwareupdate.log
- systemlog
- kernellog
- blade<#>sel

**Sample usage:**

```
wsccli# set manager tftp put -s 172.16.138.221 -f audit.log -t auditlog
```

```
wsccli# set manager tftp put -s 172.16.138.221 -f audit.log -t debuglog
```

```
wsccli# set manager tftp put -s 172.16.138.221 -f audit.log -t telemetrylog
```

```
wsccli# set manager sftp put -s 172.16.138.221 -f audit.log -t auditlog -u admin
```

**Sample output:**

Completion Code : Success

---

## 6.94 Set Manager Time

**Description:**

This sets manager system time and date in UTC.

**Syntax:**

```
wcscli# set manager time [-h] [-y YEAR] [-m MONTH] [-d DAY] [-r HOUR]
[-n MIN] [-s SECOND]
```

h, --help show this help message and exit

-y YEAR 4-digit Year

-m MONTH Numerical Month (1-12)

-d DAY Numerical Day (1-31)

-r HOUR Hour (0-23)

-n MIN Minute (1-12)

-s SEC Second (1-12)

**Sample usage:**

To change day and month, use the following command

```
wcscli# set manager time -m 9 -d 1
```

**Sample output:** Completion Code: Success

---

## 6.95 Set Manager menu On

**Description:**

This command enables the display of recovery menu that is displayed at Rack Manager U-Boot bootup

**Syntax:**

```
wcscli# set manager menu on [-h]
```

-h – Help, display the correct syntax

**Sample usage:**

```
wcscli# set manager menu on
```

**Sample output:**

---



---

Completion Code : Success

---

### 6.96 Set Manager menu Off

**Description:**

This command disables the display of recovery menu that is displayed at Rack Manager U-Boot bootup

**Syntax:**

```
wcsccli# set manager menu off [-h]
```

-h – Help, display the correct syntax

**Sample usage:**

```
wcsccli# set manager menu off
```

**Sample output:**

Completion Code : Success

---

### 6.97 Set Manager fwupdate

**Description:**

This command starts a FW update using provided firmware image file. The image file shall first be loaded to the predefined location using the tftp command get option

**Syntax:**

```
wcsccli# set manager fwupdate [-h] <-f> [-n] [-s]
```

-h – Help, display the correct syntax

-f FILE Upgrade filename or recovery image type [rollback, active, factory]

-n, Do not backup user settings - Non-default user adds,  
passwords, network config, NTP config will not be backed up

-s, Skip uboot upgrade

**Sample usage:**

```
wcsccli# set manager fwupdate <filename>
```

**Sample output:**

Completion Code : Success

Notes: When a recovery type is specified instead of a filename in the -f option, a firmware recovery operation is performed. The recovery images are stored in the unit and their versions can be obtained

---

---

using the “show manager fwupdate list” command.

---

### 6.98 Set manager defaults

**Description:**

Removes all user configuration and restores the default settings.

**Syntax:**

Wcscli# set manager defaults

**Sample usage:**

wcscli# set manager defaults

**Sample output:**

Completion Code: Success

Note: This command will reboot the unit to restore the default settings.

### 6.99 Show user info

**Description:**

Displays all the users

**Syntax:**

Wcscli# show user info [-h] -t {users,roles} [-u USERNAME]  
[-r {admin, operator, user}]

-h, --help show this help message and exit  
-t {users,roles} Provide list operations type, users or roles  
-u USERNAME User name  
-r {admin, operator, user}  
User roles: supported admin, operator, and user

**Sample usage:**

Wcscli# show user info -t roles

**Sample output:**

```
wcscli# show user info
accounts:
admin: ['root', 'demo1234', 'test098']
operator: ['op1234']
user: ['user1234', 'test321']
num_accounts: 6
```

### 6.100 Set User add

**Description:**

This command changes the OCS security role for an existing user in the manager controller.

Note that the user will be removed from other OCS security roles and added to the new role specified. Users can belong to only one security role.

**Syntax:**

```
Wcscli# set user add [-h] -u USERNAME [-p PASSWORD]
                        [-r {ocs_admin,ocs_operator,ocs_user}]
```

-h, --help        show this help message and exit

-u USERNAME       User name

-p PASSWORD       Password for user

-r {ocs\_admin,ocs\_operator,ocs\_user}

User roles: supported ocs\_admin, ocs\_operator, and ocs\_user

**Sample usage:**

To change the user role to operator, use the following command:

```
wcscli# set user add -u myname <6 characters> -p <password> -r
ocs_user
```

**Sample output:** Completion Code: Success

### 6.101 Set user update

**Description:**

This command changes the OCS security role for an existing user in the manager controller.

Note that the user will be removed from other OCS security roles and added to the new role specified. Users can belong to only one security role.

**Syntax:**

```
Wcscli# set user update [-h] -t {password,role} -u USERNAME [-p PASSWORD] [-r
{ocs_admin,ocs_operator,ocs_user}]
```

-h, --help        show this help message and exit

-t {password,role} Provide update type password or role.

-u USERNAME       User name

-p PASSWORD       Password for user

---

```
-r {ocs_admin,ocs_operator,ocs_user}
```

User roles: supported ocs\_admin, ocs\_operator, and ocs\_user

**Sample usage:**

To change the user role to operator, use the following command:

```
wcscli# set user update -t role -u myname -r ocs_operator
```

**Sample output:**

---

### 6.102 Set User Delete

**Description:**

This command removes an existing user from the manager controller.

**Syntax:**

```
Wcscli# set user delete [-h] -u USERNAME
```

-u username – Username

-h – Help, display the correct syntax

**Sample usage:**

To remove a user, use the following command:

```
wcscli# set user delete -u myname
```

**Sample output:**

### 6.103 Set manager powermeter calibration

**Description:**

This command sets the calibration values for the PRU using an input file.

**Note1:** The file has to be in the correct calibration file format. Sample file adccalibration.txt will be provided which should be used as a template.

**Note2:** Calibration values from the file will be programmed both in the EEPROM (for persistence) as well as in the PRU. Frequent execution of this command should be avoided since it will wear out the EEPROM.

**Syntax:**

```
WcsCli# set manager powermeter calibration -h
```

usage: set manager powermeter calibration [-h] -f FILENAME

optional arguments:

-h, --help show this help message and exit

---

---

-f FILENAME calibration file path (syntax must match)

**Sample Usage:**

Wcscli# set manager powermeter calibration -f adccalibration.txt  
Completion Code: Success

#### 6.104 Set manager throttle force enable

**Description:**

This command enables or disables the throttle signal.

**Syntax:**

WcsCli# set manager throttle force -h  
usage: set manager throttle force [-h] {enable} ...

positional arguments:

{enable} Force throttle action  
enable This command enables force throttle signal

**Sample usage:**

Disable force throttling:

WcsCli# set manager throttle force enable -e 0  
Completion Code: Success

Enable force throttling:

WcsCli# set manager throttle force enable -e 1  
Completion Code: Success

#### 6.105 Set manager throttle count clear

**Description:**

This command clears the throttle count and reset the last throttle count rest time.

**Syntax:**

WcsCli# set manager throttle count -h  
usage: set manager throttle count [-h] {clear} ...

positional arguments:

{clear} throttle count clear action  
clear This command clears throttle count

**Sample usage:**

WcsCli# set manager throttle count clear

---

---

Completion Code: Success

### 6.106 Set manager pubsub config

**Description:**

This command add/update PubSub configs(Brokers, Topic, and Ssl) settings.

**Syntax:**

WcsCli# set manager pubsub -h

usage: set manager pubsub [-h] [-l BROKERS] [-t TOPIC] [-s SSL]

optional arguments:

- h, --help show this help message and exit
- l BROKERS PubSub broker IP addresses
- t TOPIC PubSub topic name
- s SSL PubSub SSL authentication enable

**Sample usage:**

WcsCli# set manager pubsub -l 10.10.10.500 -t CSIRM -s 0

Completion Code: Success

### 6.107 Set manager userconfigs

**Description:**

This command set the userconfig “keyname” to the value “value”

**Syntax:**

- WcsCli# set manager userconfigs -h
- usage: set manager userconfigs [-h] -k KEYNAME -v VALUE
- 
- optional arguments:
- -h, --help show this help message and exit
- -k KEYNAME User configs key to set
- -v VALUE User configs value to set

**Sample usage:**

WcsCli# set manager userconfigs -k PowercapMultiPolicyEnable -v 1

Completion Code: Success

### 6.108 Set manager ssh config

**Description:**

This command set the ssh default or advanced cipher security level

**Syntax:**

- WcsCli# set manager ssh config -h
- usage: set manager ssh config [-h] -s SECURITYLEVEL
- 
- optional arguments:
- -h, --help show this help message and exit
- -s SECURITYLEVEL set ssh cipher precedence level

**Sample usage:**

WcsCli# set manager ssh config -s 1  
Completion Code: Success

## 7 Server Commands

### *Show Commands*

- [show system info](#)
- [show system health](#)
- [show system fru](#)
- [show system state](#)
- [show system presence](#)
- [show system default Power](#)
- [show system power Limit](#)
- [show system power reading](#)
- [show system power Alert](#)
- [show system power Throttle](#)
- [show system fpga health](#)
- [show system fpga mode](#)
- [show system fpga temp](#)
- [show system fpga i2cversion](#)
- [Show system fpga assetinfo](#)
- [show system boot](#)
- [show system Log read](#)
- [show system bios config](#)
- [show system bios code](#)
- [show system bios update](#)
- [show system tpm Presence](#)
- [show system psu info](#)
- [show system psu battery](#)
- [show system psu update](#)
- [show system psu version](#)
- [Show system type](#)
- [show system datasafe policy](#)
- [show system datasafe trigger](#)
- [show system nvme](#)
- [show system led](#)
- [show system file list](#)

### *Set Commands*

- [set system log clear](#)
- [set system power alert](#)



- [Set system tpm presence](#)
- [set system on](#)
- [set system off](#)
- [set system reset](#)
- [set system led on](#)
- [set system led off](#)
- [set system default power on](#)
- [set system default power off](#)
- [set system power limit value](#)
- [set system power limit on](#)
- [set system power limit off](#)
- [set system nextboot](#)
- [set system bios config](#)
- [set system bios update](#)
- [set system fpga bypass enable](#)
- [set system fpga bypass disable](#)
- [set system psu clear](#)
- [set system psu battery](#)
- [set system psu update](#)
- [set system rmedia mount](#)
- [set system rmedia unmount](#)
- [set system file delete](#)
- [set system file get](#)
- [set system file put](#)

#### *Serial Commands*

- [Start Serial Session](#)
- [Stop Serial Session](#)
- [Start jtag service](#)
- [Stop jtag service](#)

**Description:**

This command shows information about the system, including serial number and version information.

**Syntax:**

```
Wcscli# show/sh system info <-i server_index > [-h]
```

-i – Server index (1-48)

-h – Help, display the correct syntax

**Sample usage:**

To get information for system 1, execute the following command:

```
Wcscli# show system info -i 1
```

**Sample output:**

```
AssetTag: N/A
  Boot_BootSourceOverrideEnabled: False
  Boot_BootSourceOverrideTarget: No override
  ChassisId: ChassisId
  Description: Builtin FRU Device (ID 0)
  HostName: 172.17.0.117
  Id: 30
  Manufacturer: Microsoft
  Model: N/A
  Name: Server30
  PartNumber: X943064-005
  SKU: N/A
  SerialNumber: B5500T01D0116340014EJ0LC
  Server:
    BMCVersion: C2010.BC.02.06.111
    BiosVersion: C2010.BS.1E02.GN1
    CpIdVersion: 00000010
    HWVersion: C2012
    IndicatorLED: Off
    MemorySummary:
      Status:
        Health: Ok
        HealthRollUp: Ok
        State: Enabled
      TotalSystemMemoryGib: 64
    PowerState: On
    ProcessorSummary:
      Count: 2
      Model: Skylake
      Status:
        Health: Ok
        HealthRollUp: Ok
        State: Present
```

```
Status:
```

```
Health: Ok
```

## 7.2 Show system fru

### Description:

This command gets the FRU information of the specified server.

### Syntax:

```
Wcscli# show system fru -i <serverId>
```

-i – Server index (1-48)

-h – Help, display the correct syntax

### Sample usage:

To get the asset (FRU) information for server 1, use the following command:

```
Wcscli# show system fru -i 1
```

### Sample output:

AssetTag: N/A

Board Mfg: Microsoft

Board Mfg Date: Tue Aug 30 08:19:00 2016

Board Part Number: X943064-005

Board Product: C2012

Board Serial Number: B5500T01D0116340014EJ0LC

Manufacturer: Microsoft

Model: N/A

Product Part Number: M1011544-001

Product Serial: B8100T01003063500007J0SA

Product Version: 4.0

Completion Code: Success

## 7.3 Show system State

### Description:

This command gets the ON/OFF state of the server (whether the server chipset is receiving power).

- When ON, server is receiving AC power (hard power state) and the chipset is receiving power

(soft power state).

- When OFF, server chipset is not receiving power.

This command also returns the status of the data save status (NVDIMM save and non-plp PCIe save) using “DatasaveStatus” and the amount of time in seconds to wait for save operation to complete using “DatasaveRemainingTime” (when save is pending).

Possible values for DatasaveStatus are,

Unknown  
NotApplicable  
SaveNotStarted  
SavePending  
SaveCompleted

DatasaveRemainingTime in seconds is reported when DatasaveStatus is SavePending

**Syntax:**

```
Wcscli# show system <-i server_index> [-h] [-b]
```

-i – Server index (1-48)

-h – Help, display the correct syntax

-b DEVICEID device id (0:Host OS or 1:Soc), by default 0.

**Sample usage:**

To get the ON/OFF state of server 1, use the following command:

```
WcsCli# show system state -i 2
  DatasaveStatus: SaveNotStarted
  State: ON
  Completion Code: Success
```

When save is pending,

```
WcsCli# show system state -i 37 -b 1
  State: ON
  Completion Code: Success
```

```
WcsCli# show system state -i 2
  DatasaveRemainingTime: 365
  DatasaveStatus: SavePending
  State: ON
  Completion Code: Success
```

---

## 7.4 Show System Presence

**Description:**

This command shows the system physical presence

**Syntax:**

```
Wsccli# show system presence <-i server_index > [-h]
```

-i – server index (1-48)

-h – Help, display the correct syntax

**Sample usage:**

To get the power state for server 1, use the following command:

```
wsccli# show system presence -i 1
```

**Sample output:**

```
Port Presence: True
```

```
Completion Code: Success
```

---

## 7.5 Show System Default Power

**Description:**

This command gets the default power state of the server ON/OFF.

**Syntax:**

```
Wsccli# show system default power <-i server_index > [-h]
```

-i – server index, the number of servers (1-48)

-h – Help, display the correct syntax

**Sample usage:**

To set the default power state of of server 1 to ON, use the following command:

```
wsccli# show system default power -i 1
```

**Sample output:**

```
Default Power State: OFF
```

```
Completion Code: Success
```

---

## 7.6 Show System Power Limit

**Description:**

This command shows the power limit for a server.

**Syntax:**

---

---

```
Wcscli# show system power limit <-i server_index> [-h]
```

-i – server index, the number of servers (1-48)

-h – Help, display the correct syntax

**Sample usage:**

To show the power limit for server 1, use the following command:

```
Wcscli# show system power limit -i 1
```

**Sample output:**

StaticLimit: 740 Watts

StaticState: No Active Power Limit

Completion Code: Success

---

## 7.7 Show System Power Reading

**Description:**

This command gets the power reading for a server. The command can be used for monitoring and or other power control mechanism (refer to the **SetSystemPowerLimit** command).

**Syntax:**

```
Wcscli# show system power reading <-i server_index > [-h]
```

-i – Server index, the number of servers (1-48)

-h – Help, display the correct syntax

**Sample usage:**

To show the power reading for server 1, use the following command:

```
wcscli# show system power reading -i 1
```

**Sample output:**

power reading: 67 Watts

---

## 7.8 Show System Power Alert

**Description:**

This command gets the power alert policy for a system

**Syntax:**

```
Wcscli# show system power alert <-i server_index > [-h]
```

-i – Server index, the number of servers (1-48)

-h – Help, display the correct syntax

**Sample usage:**

To show the power alert policy for server 1, use the following command:

---

---

```
wsccli# show system power alert -i 1
```

**Sample output:**

```
AutoRemovePowerLimitDelayInSecs : 20 ms
Completion Code : Success
FastThrottleDurationInMsecs : 0 sec
IsServerPowerAlertLimitEnabled : False
IsServerPowerAlertLimitStatus : 16
Server ID : 22
ServerAlertAction : NoAction
ServerPowerAlertLimitInWatts : 250 Watts
ServerPowerDrawnInWatts : 0 Watts
```

---

## 7.9 Show System Power Throttle

**Description:**

This command gets the throttling statistics for a System

**Syntax:**

```
wsccli# show system power throttle [-h] -i system-id
-i – Server index, the number of servers (1-48)
-h – Help, display the correct syntax
```

**Sample usage:**

To show the power throttling statistics for server 1, use the following command:

```
wsccli# show system power throttle -i 1
```

**Sample output:**

```
Completion Code : Success
IsServerPowerAlertLimitStatus : 16
Server ID : 22
ServerPowerAlertLimitInWatts : 250 Watts
ServerPowerDrawnInWatts : 0 Watts
```

---

## 7.10 Show System FPGA Health

**Description:**

This command gets the system FPGA health

**Syntax:**

```
wsccli# show system fpga health [-h] -i system-id
-i – Server index, the number of servers (1-48)
-h – Help, display the correct syntax
```

**Sample usage:**

---

To show the fpga health for server 1, use the following command:

```
wsccli# show system fpga health -i 1
```

**Sample output:**

```
40G Link 0 Rx Activity: 0
40G Link 0 Tx Activity: 0
40G Link 0 Up: 0
40G Link 1 Rx Activity: 0
40G Link 1 Tx Activity: 0
40G Link 1 Up: 0
PCIe HIP 0 Up: 1
PCIe HIP 1 Up: 1
Completion Code: Success
```

---

### 7.11 Show System FPGA Mode

**Description:**

This command gets the system FPGA bypass mode setting

**Syntax:**

```
wsccli# show system fpga mode [-h] -i system-id
-i – Server index, the number of servers (1-48)
-h – Help, display the correct syntax
```

**Sample usage:**

To show the fpga mode for server 1, use the following command:

```
wsccli# show system fpga mode -i 1
```

**Sample output:**

```
Bypass Mode: Off
User Logic Network: Disabled
Completion Code: Success
```

---

### 7.12 Show System FPGA Temp

**Description:**

This command gets the system FPGA temperature in Celsius

**Syntax:**

```
wsccli# show system fpga temp [-h] -i system-id
-i – Server index, the number of servers (1-48)
```

---



---

-h – Help, display the correct syntax

**Sample usage:**

To show the fpga temp for server 1, use the following command:

```
wsccli# show system fpga temp -i 1
```

**Sample output:**

```
Temperature in Celsius: 44  
Completion Code: Success
```

---

### 7.13 Show System FPGA I2CVersion

**Description:**

This command gets the system FPGA I2C version

**Syntax:**

```
Wsccli# show system fpga i2cversion [-h] -i system-id  
-i – Server index, the number of servers (1-48)  
-h – Help, display the correct syntax
```

**Sample usage:**

To show the fpga i2c version for server 1, use the following command:

```
wsccli# show system fpga i2cversion -i 1
```

**Sample output:**

```
I2C Version: 0  
Completion Code: Success
```

---

### 7.14 Show System FPGA AssetInfo

**Description:**

This command gets the system FPGA asset information

**Syntax:**

```
Wsccli# show system fpga assetinfo [-h] -i system-id  
-i – Server index, the number of servers (1-48)  
-h – Help, display the correct syntax
```

**Sample usage:**

To show the fpga assetinfo for server 1, use the following command:

```
wsccli# show system fpga assetinfo -i 1
```

---

**Sample output:**

```
Product Manufacturer:
Product Name:
Product Model Number:
Product Version:
Product Serial Number:
Product FRU File ID:
Product Custom Field 1:
Product Custom Field 2:
Completion Code: Success
```

---

### 7.15 Show System Boot

**Description:**

This command gets the pending boot order to be applied the next time the server boots.

Once the boot order is applied, GetNextBoot will return the default value of NoOverRide.

**Syntax:**

```
WcsCli# show system boot -h
usage: show system boot [-h] [-b {0,1}] -i server-id
```

optional arguments:

```
-h, --help      show this help message and exit
-b {0,1}        device id (0:Host OS or 1:Soc)
```

required arguments:

```
-i server-id    server-id
```

**Sample usage:**

To get the next boot device type for server 1, use the following command:

```
wcscli# show system boot -i 21
```

**Sample output:**

```
BootSourceOverrideEnabled: Once
BootSourceOverrideMode: Legacy
BootSourceOverrideTarget: Force PXE
Next boot is: Force PXE
Completion Code: Success
```

---

### 7.16 Show system Log read

**Description:**

This command reads the log from a server. The server log (system event log) contains information

---

about events/warnings/alerts pertaining to that server like thermal throttling of servers due to overheating, etc.

**Syntax:**

```
wsccli# show system log read <-i server_index> [-h]
```

- i – Server index, the number of servers (1-24)
- h – Help, display the correct syntax

**Sample usage:**

To read the server log for server 1, use the following command:

```
wsccli# show system log read -i 21
```

**Sample output:**

1d		11/09/2016		19:49:13		Event Logging Disabled SEL   Log area reset/clear		ed   Asserted
1e		11/09/2016		19:49:30		Power Supply PSU_Status_Event   Power Supply AC 1		ost   Asserted
1f		11/09/2016		19:49:30		Power Supply PSU_Status_Event   Power Supply AC 1		ost   Asserted
20		11/09/2016		19:49:31		Power Supply PSU_Status_Event   Power Supply AC 1		ost   Deasserted
21		11/09/2016		19:49:31		Power Supply PSU_Status_Event   Power Supply AC 1		ost   Deasserted
22		11/09/2016		22:39:36		Unknown NM SmarTCLST   State Deasserted   Asserted		

7.17 Show Bios Config

**Description:**

GetServerBiosConfiguration API returns the following information corresponding to the server’s BIOS configuration,

1. BiosConfiguration currentConfig
  - ‘currentConfig’ specifies the Bios configuration setting that is currently active in the server.
2. BiosConfiguration chosenConfig
  - ‘chosenConfig’ specifies the Bios configuration setting that is chosen by the user (via the SetServerBiosConfiguration API) and will be activated in the subsequent server reboot operation.
3. List<AvailableBiosConfiguration> availableConfigs
  - ‘availableConfigs’ is the list of BIOS configurations that are supported in the BIOS. The user can review the list and choose the appropriate configuration for the server via the SetServerBiosConfiguration API.

‘BiosConfiguration’ is a 4 value tuple that uniquely identify a certain configuration in the BIOS,

- Int majorConfig

SW: RM CLI

- Specifies the major configuration identifier (see AvailableBiosConfiguration below)
- Int minorConfig
  - Specifies the minor configuration identifier (see AvailableBiosConfiguration below)

‘AvailableBiosConfiguration’ list the available Bios configurations and provides a semantic association to the configuration in the form of a 3-character ‘ConfigName’. Example of ‘ConfigName’ are, GN1, AZ1, BN1, BN2 etc.

**Syntax:**

```
wcscli# show system bios config -i <server-index>
-i - server Id
-h - help; display the correct syntax
```

**Sample usage:**

```
wcscli# show system bios config -i 21
```

**Sample output:**

```
Current BIOS Configuration : 0.0
Chosen BIOS Configuration : 1.2
AvailableConfigName : GN1
```

7.18 Show BIOS Code

**Description:**

This command gets the post code for the server with specified server id.

**Syntax:**

```
wcscli# show system bios code <-i server_index> [-h]
server_index - the target server number. Typically 1-48
-h - help; display the correct syntax
```

**Sample usage:**

```
wcscli# show system bios code -i 27
```

**Sample output:**

```
Bios Code : [' b7 bd bd bd bd b7 bd bd bd bd b7 bd bd bd bd b7 bd bd bd bd b7 b7 bd bd bd bd bd bd
bd bd bd b7 bd bd bd bd bd bd bd bd bd b7 bd bd bd bd bd bd bd bd bd bd bd bd bd bd bd bd bd bd
b7 bd bd bd bd bd bd bd bd bd bd bd bd bd bd bd bd bd bd bd bd b7 b7 b7 b7 b7 b7 bd bd b7 b7 bd bd b7
be b7 b7 b7 b7 b7 b7 b7 b8 b8 b8 b8 b8 b8 b8 b8 b9 b9 b9 b9 b9 b9 b9 b9 ba b9 b9 bb bb bb bb bb
```

SW: RM CLI

---

```
bb bb bb bb bb bb b7 bc bc bc bc bc bc bc bc bf e6 e7 e8 e9 eb ec ed ee e7 e8 e9 eb ec ed ee 4f
61 9a 78 68 70 79 d0 d1 d2 d3 d4 91 d5 92 94 95 94 94 95 94 94 94 94 94 95 94 94 94 95 94 95 94
95 94 95 94 94 95 96 ef 92 92 92 92 92 92 97 b2 99 98 9d 9c b4 b4 b4 92 a0 a2 a2 a2 a2 a0 a2 a2 a2
a2 a2 a2 a2 a2 99 92 92 92 92 92 92 92 b2', "]
```

---

### 7.19 Show system bios update

This command shows BIOS firmware update status.

#### Syntax:

```
wcscli# show system bios update [-h] -i <server-id>
```

server\_index - the target server number. Typically 1-48

-h - help; display the correct syntax

#### Sample usage:

```
wcscli# show system bios update -i 30
```

#### Sample output:

```
FW Status: FW update not started
Completion Code: Success
```

---

### 7.20 Show system bmc update

This command shows BMC firmware update status.

#### Syntax:

```
wcscli# show system bmc update [-h] -i <server-id>
```

server\_index - the target server number. Typically 1-48

-h - help; display the correct syntax

#### Sample usage:

```
wcscli# show system bmc update -i 30
```

#### Sample output:

```
FW Status: FW update not started
Completion Code: Success
```

---

### 7.21 Show system cpld update

This command shows CPLD firmware update status.

---

**Syntax:**

```
wcscli# show system cpld update [-h] -i <server-id>
```

server\_index - the target server number. Typically 1-48

-h - help; display the correct syntax

**Sample usage:**

```
wcscli# show system cpld update -i 30
```

**Sample output:**

FW Status: FW update not started

Completion Code: Success

---

## 7.22 Show Tpm Physical Presence

**Description:**

This command checks whether TPM physical presence is set

**Syntax:**

```
wcscli# show system tpm presence [-h] -i <server-id>
```

server\_index - the target server number. Typically 1-48

-h - help; display the correct syntax

**Sample usage:**

```
wcscli# show system tpm presence -i 27
```

**Sample output:**

PhysicalPresence : False

Completion Code: Success

---

## 7.23 Show System Psu Info

**Description:**

This command reads the PSU info

**Syntax:**

```
wcscli# show system psu Info [-h] -i <server-id> [-p {0,1,2}]
```

- i server\_index - the target server number. Typically 1-48
- p – Phases - psu info with phase options default shows all phases details
- h - help; display the correct syntax

**Sample usage:**

```
wsccli# show system psu info -i 27
wsccli# show system psu info -i 27 -p 1
```

**Sample output:**

[Default all phases]

Battery: No  
Capability: 0xb0  
Device ID:  
Faults:  
Iin: 0.968750A  
Iout: 16.875000A  
Manufacturer:  
Model: --BANFF-900W-01  
PSU Feed: B  
Pin: 213.000000W  
Pout: 207.000000W  
Serial: 5WBYC03NF300HZ0  
Status Word: 0x0000  
Vin: 211.000000V  
Completion Code: Success

**[Phase 1 Output]**

Faults:  
Iin: 0.312500A  
Pin: 67.000000W  
Status Word: 0x0000  
Vin: 211.000000V  
Completion Code: Success

## 7.24 Show system psu battery

This command shows psu battery presence.

**Syntax:**

```
wsccli# show system psu battery [-h] -i <server-id>
```

server\_index - the target server number. Typically 1-48

-h - help; display the correct syntax

**Sample usage:**

```
wcscli# show system psu battery -i 27
```

**Sample output:**

```
Battery: No  
Capability: 0xb0  
Completion Code : Success
```

---

## 7.25 Show system psu update

This command shows psu firmware update status.

**Syntax:**

```
wcscli# show system psu update [-h] -i <server-id>
```

server\_index - the target server number. Typically 1-48

-h - help; display the correct syntax

**Sample usage:**

```
wcscli# show system psu update -i 30
```

**Sample output:**

```
FW Status: FW update not started  
Completion Code: Success
```

---

## 7.26 Show system psu version

This command shows psu firmware version.

**Syntax:**

```
wcscli# show system psu version [-h] -i <server-id> [-v {fw, bootloader}]
```

server\_index - the target server number. Typically 1-48

-v version optional argument {fw or boot loader} default fw

-h - help; display the correct syntax

**Sample usage:**

```
wcscli# show system psu version -i 30
```

**Sample output:**

```
Wcscli# show system psu version -i 30 [default returns fw version]  
FW Active: Yes
```



---

```
FW Image: B
FW Revision: X4.06.04
Completion Code: Success
```

```
wcscli# show system psu version -i 30 -v bootloader
Bootloader FW Active: Yes
Bootloader FW Image: Bootloader
Bootloader FW Revision: 01.02.02
Completion Code: Success
```

---

### 7.27 Show system nvme

This command shows nvme status.

**Syntax:**

```
wcscli# show system nvme [-h] -i <server-id>
```

server\_index - the target server number. Typically 1-48

-h - help; display the correct syntax

**Sample usage:**

```
wcscli# show system nvme -i 30
```

**Sample output:**

```
Wcscli# show system nvme -i 30
```

```
wcscli# show system nvme -i 30
```

```
Port Type:
Slot:
Device:
Vendor:
Temp:
Drive Status:
Drive Presence:
SMBUS Arbitration:
Drive Not Read:
Drive Functional:
No Reset Required:
Port1 Link Active:
Port2 Link Active:
SMART Warn:
Serial No:
Completion Code: Success
```

---

---

### 7.28 Show System Type

This command shows system type.

**Syntax:**

```
wcscli# show system type [-h] -i <server-id>
```

server\_index - the target server number. Typically 1-48

-h - help; display the correct syntax

**Sample usage:**

```
wcscli# show system type -i 30
```

**Sample output:**

```
Wcscli# show system type -i 30
Control Interface Type: IPMI
OEM ID: 311
System Type: C2010
Completion Code: Success
```

---

### 7.29 Show System datasafe policy

This command show datasafe (NVDIMM, PCIe) policy settings

**Syntax:**

```
wcscli# show system datasafe policy [-h] -i <server-id>
```

server\_index - the target server number. Typically 1-48

-h - help; display the correct syntax

**Sample usage:**

```
wcscli# show system datasafe policy -i 30
```

**Sample output:**

```
Wcscli# show system datasafe policy -i 30
NVDIMM Backup Delay: 0
PCIe Reset Delay: 2
Completion Code: Success
```

---

### 7.30 Show System datasafe trigger

This command shows datasafe trigger status

---

**Syntax:**

```
wcscli# show system datasafe trigger [-h] -i <server-id>
```

server\_index - the target server number. Typically 1-48

-h - help; display the correct syntax

**Sample usage:**

```
wcscli# show system datasafe trigger -i 30
```

**Sample output:**

```
Wcscli# show system datasafe trigger -i 30
NVDIMM Backup Delay: 0
PCIe Reset Delay: 2
Completion Code: Success
```

---

### 7.31 Show System led

This command shows system led state.

**Syntax:**

```
wcscli# show system led [-h] -i <server-id>
```

server\_index - the target server number. Typically 1-48

-h - help; display the correct syntax

**Sample usage:**

```
wcscli# show system led -i 30
```

**Sample output:**

```
Wcscli# show system type -i 30
LED Status: Off
Completion Code: Success
```

---

### 7.32 Show System File List

**Description:**

This command list files in the server available for file transfer operations

**Syntax:**

```
Wcscli# show system file list [-h] -i server-id
```

-i – server index, the number of servers (1-48)

-h – Help, display the correct syntax

**Sample usage:**

---

```
wsccli# show system file list -i 1
```

**Sample output:**  
FTPFiles:  
Completion Code: Success

7.33 Set Log Clear

**Description:**  
This command clears the logs from a server.

**Syntax:**  
Wsccli# set system log clear [-h] -i server-id  
-i – server index, the number of servers (1-48)  
-h – Help, display the correct syntax

**Sample usage:**  
To clear the server log for server 1, use the following command:  
wsccli# set system log clear -i 1

**Sample output:**

7.34 Set Power Alert

**Description:**  
This command sets the power alert policy for a system

**Syntax:**  
Wsccli# set system power alert [-h] -i system-id -p [120-1000] -e  
alert-action -r remediation-action  
[-f throttle-duration] [-d remove-  
delay]  
-i system-id system-id (1 to 48)  
-p [120-1000] power limit in watts  
-e alert-action alert action, 0:nothing, 1:throttle, 2:fast  
throttle  
-r remediation-action remediation action, 0:nothing, 1:remove limit  
and  
aearm alert, 2:rearm alert  
-f throttle-duration fast throttle duration in milliseconds  
-d remove-delay auto remove power limit delay in seconds

**Sample**

SW: RM CLI

**usage:**

To show the power alert policy for server 1, use the following command:

```
wcscli# set system power alert -i 1
```

**Sample output:**

---

### 7.35 Set System On

**Description:**

This command supplies the power to the server chipset, initializing the boot process. This command is used to soft power the server ON.

This command is an asynchronous command - the command will return immediately but it may take from few seconds to few minutes to complete due to potential data save operation (NVDIMM or non-PLP PCIe save).

The command succeeds when there is no pending save operation (from a previously initiated system off/reset operation) on the blade. The command fails when there is pending save operation. The returned failure message (StatusDescription) will include the amount of time to wait for save operation to complete, at which point this command may be re-attempted. The user may also use "show system state" command to query the status of the save operation.

**Syntax:**

```
wcscli# set system on <-i server_index | -a> [-h]
```

Optional arguments:

- h, --help show this help message and exit
- b DEVICEID device id (0:Host OS or 1:Soc), by default 0

Required arguments:

- i SERVERID server-id

**Sample usage:**

To soft power ON server 1, use the following command:

```
wcscli# set system on -i 1
```

**Sample output:**

When save is not pending,

```
WcsCli# set system on -i 2
```

```
State: ON
```

```
Completion Code: Success
```

When save is pending,

```
WcsCli# set system on -i 2
```

Status Description: Operation not permitted for blade (2) due to pending save, retry after (365) seconds

Completion Code: Failure

---

### 7.36 Set System Off

#### Description:

This command removes the power from the server chipset. This command is used to soft power the server OFF.

This command is an asynchronous command - the command will return immediately but it may take from few seconds to few minutes to complete due to potential data save operation (NVDIMM or non-PLP PCIe save).

The command succeeds when there is no pending save operation (from a previously initiated system off/reset operation) on the blade. The command fails when there is pending save operation. The returned failure message (StatusDescription) will include the amount of time to wait for save operation to complete, at which point this command may be re-attempted. The user may also use "show system state" command to query the status of the save operation.

#### Syntax:

```
wsccli# set system off <-i server_index | -a> [-h]
```

Optional arguments:

- h, --help show this help message and exit
- b DEVICEID device id (0:Host OS or 1:Soc), by default 0

Required arguments:

- i SERVERID server-id

#### Sample usage:

To soft power OFF server 1, use the following command:

```
wsccli# set system off -i 1
```

#### Sample output:

When save is not pending (this command will return success and initiate save),

```
WcsCli# set system off -i 2
State: OFF
Completion Code: Success
```

When save is pending,

```
WcsCli# set system off -i 2
```

---

---

```
Status Description: Operation not permitted for blade (2) due to
pending save, retry after (365) seconds
Completion Code: Failure
```

---

### 7.37 Set System Reset

**Description:**

This command power cycles or soft resets the server(s).

Power cycle resets the server (causing a software reboot sequence). Server AC power signal remains ON throughout this process. Any serial session active on that server will continue to be active during this process.

This command is an asynchronous command - the command will return immediately but it may take from few seconds to few minutes to complete due to potential data save operation (NVDIMM or non-PLP PCIe save).

The command succeeds when there is no pending save operation (from a previously initiated system off/reset operation) on the blade. The command fails when there is pending save operation. The returned failure message (StatusDescription) will include the amount of time to wait for save operation to complete, at which point this command may be re-attempted. The user may also use "show system state" command to query the status of the save operation.

**Syntax:**

```
wsccli# set system reset [-h] -i server-id
```

Optional arguments:

```
-h, --help    show this help message and exit
-b DEVICEID  device id (0:Host OS or 1:Soc), by default 0.
```

Required arguments:

```
-i SERVERID  server-id
```

**Sample usage:**

To power cycle server 3, use the following command:

```
wsccli# set system reset -i 3
```

To power cycle server 37 Soc, use the following command:

```
wsccli# set system reset -i 37 -b 1
```

**Sample output:**

When save is not pending (this command will succeed and initiate save),

```
WcsCli# set system reset -i 2
```

---

```
Completion Code : Success
```

```
When save is pending,
```

```
WcsCli# set system reset -i 2
```

```
    Status Description: Operation not permitted for blade (2) due to  
    pending save, retry after (365) seconds
```

```
    Completion Code: Failure
```

```
WcsCli# set system reset -i 37 -b 1
```

```
Completion Code : Success
```

---

### 7.38 Set System LED On

**Description:**

This command turns the server UID LED On. The purpose of this UID LED is to identify the server.

**Syntax:**

```
Wcscli# set system led on <-i server_index> [-h]
```

-i – server index, the number of servers (1-48)

-h – Help, display the correct syntax

**Sample usage:**

To turn the server UID LED for server 1 ON, use the following command:

```
wcscli# set system led on -i 1
```

**Sample output:**

```
Completion Code: Success
```

---

### 7.39 Set System LED Off

**Description:**

This command turns the server UID LED Off. The purpose of this UID LED is to identify the server.

**Syntax:**

```
Wcscli# set system led off <-i server_index> [-h]
```

-i – server index, the number of servers (1-48)

-h – Help, display the correct syntax

**Sample usage:**

To turn the server UID LED for server 1 OFF, use the following command:

```
wcscli# set system led off -i 1
```

---



**Sample output:**

Completion Code: Success

**7.40 Set System Default Power State On**

**Description:**

This command sets the default power state of the server ON.

The default server power state denotes the behavior of the server after receiving AC power, either when a server is initially inserted in to its slot or power returns after a utility failure. If the server default power state is set to OFF, the server won't be powered ON after receiving AC input power. An explicit **SetServerOn** command needs to be sent to power ON the server. If the server default power state is set to ON, the server will be powered ON after receiving AC input power.

Note that the server default power state does not affect the active power state of the server, only their behaviors after a hard power recycle.

**Syntax:**

```
wcscli# set system default power on <-i server_index > [-h]
-i - Server index (1-48)
-h - Help, display the correct syntax
```

**Sample usage:**

To set the default power state of of server 1 to ON, use the following command:

```
wcscli# set system default power on -i 23
```

**Sample output:**

```
Completion Code : Success
Default Power State : ON
```

**7.41 Set System Default Power State Off**

**Description:**

This command sets the default power state of the server OFF.

**Syntax:**

```
wcscli# set system default power off <-i server_index > [-h]
-i - Server index (1-48)
-h - Help, display the correct syntax
```

**Sample usage:**

---

To set the default power state of of server 1 to OFF, use the following command:

```
wsccli# set system default power off -i 23
```

**Sample output:**

```
Default Power State : OFF
```

```
Completion Code : Success
```

---

---

## 7.42 Set System Power Limit Value

**Description:**

This command sets the power limit for a server.

The power limit can be set for a variety of reasons including energy savings and under provisioning (consolidate more servers under a power hierarchy).

**Syntax:**

```
Wsccli# set system power limit value <-i server_index> <-l power_limit> [-h]
```

-i – Server index, the number of servers (1-48)

-l – Server power limit, in W

-h – Help, display the correct syntax

**Sample usage:**

To set the power limit for server 1 to 750 W, use the following command:

```
Wsccli# set system power limit value -i 1 -l 500
```

**Sample output:**

```
Completion Code : Success
```

---

---

### 7.43 Set System Power Limit On

**Description:**

This command activates the power limit for a server, and enables power throttling.

**Syntax:**

```
wcscli# set system power limit on <-i server_index> [-h]
```

-i – Server index, the number of servers (1-48)

-h – Help, display the correct syntax

**Sample usage:**

To activate the power limit for server 1, use the following command:

```
wcscli# set system power limit on -i 1
```

**Sample output:**

```
Completion Code: Success
```

### 7.44 Set System Power Limit Off

**Description:**

This command deactivates the power limit for a server, and disables power throttling.

**Syntax:**

```
wcscli# set system power limit off <-i server_index > [-h]
```

-i – Server index, the number of servers (1-48)

-h – Help, display the correct syntax

**Sample usage:**

To deactivate the power limit for server 1, use the following command:

```
wcscli# set system power limit off -i 1
```

**Sample output:**

```
Completion Code: Success
```

## 7.45 Set system Boot

### Description:

This command sets the device boot type of the start boot device during the subsequent reboot for a server. A soft power cycle (setserveractivepowercycle or setserveroff, then setserveron) after setting the boot options is required for the boot setting to take effect.

### Syntax:

**WcsCli# set system boot -h**

```
usage: set system boot [-h] [-b {0,1}] [-m boot-mode] [-p is-persistent]
-i
                        SERVERID -t boot-type
```

### optional arguments:

```
-h, --help            show this help message and exit
-b {0,1}              device id (0:Host OS or 1:Soc)
-m boot-mode          Boot mode 0- Legacy or 1- UEFI
-p is-persistent      0- Non-persistent or 1- Persistent
```

### required arguments:

```
-i SERVERID          server-id
-t boot-type          boot-type: 1. none: No override ;
                        2. pxe:Force PXE boot;
                        3. disk: Force boot from default Hard-drive;
                        4. bios:Force boot into BIOS Setup;
                        5. remotedriver: Force boot from remote drive -
                           virtual USB
                        6. emmc: Force boot from eMMC
                        7. usb: Force boot from attached USB
```

```
-m bootmode : 0 Legacy 1 UEFI
```

```
-p Persistent : 0 Non-persistent 1 persistent
```

### Sample usage:

To set the boot device for server 1, use the following command:

```
wcscli# set system boot -i 1 -t pxe
```

### Sample output:

```
Completion Code : Success
Next boot : Set Boot Device to pxe
```

---

## 7.46 Set System Bios Config

### Description:

SetServerBiosConfiguration sets the configuration that the users expects in the subsequent reboot of the server. We expect the user to use the GetServerBiosConfiguration API and review the AvailableBiosConfiguration list and select the chosen configuration which can be set with this API.

### Syntax:

```
Wcscli# set system bios config <-i server_id> <-j majorConfig> <-n minorConfig> [-h]
-i - server Id
-j - major configuration number (refer to getServerBiosConfig available configurations)
-n - minor configuration number (refer to getServerBiosConfig available configurations)
-h - help; display the correct syntax
```

### Sample usage:

usage: set system bios config [-h] -i server-id -j major-config -n minor-config

### Sample output:

Completion Code : Success

---

## 7.47 Set system psu clear

### Description:

This command clears the psu faults

### Syntax:

```
Wcscli# set system psu clear <-i server_id> -p {0,1,2} phases (0 {phase 1}, 1 {phase 2}, 2 {phase 3},
{default all phases})
```

### Sample usage:

usage: set system psu clear [-h] -i server-id

-i <Server-id> Serever id <1 to 48 ids>  
-p <Phases> phases (0 {phase 1}, 1 {phase 2}, 2 {phase 3}, {default all phases})

### Sample output:

Completion Code : Success

---

#### 7.48 Set system psu battery

**Description:**

This command runs the battery health test on the specified PSU.

**Syntax:**

Wcscli# set system psu battery <-i server\_id>

**Sample usage:**

usage: set system psu battery [-h] -i server-id

-i <Server-id> Serever id <1 to 48 ids>

**Sample output:**

Completion Code : Success

#### 7.49 Set system psu update

**Description:**

The firmware update for the server PSU firmware

**Syntax:**

Wcscli# set system psu update [-h] -i server-id -f <file-path> -t {0,1,2}

**Sample usage:**

usage: set system psu update [-h] -i server-id -f </tftp/psufw.txt> -t 1

-i <Server-id> Serever id <1 to 48 ids>

-f file-path firmware file path to update.

-t Type of the firmware image {0: bootloader , 1:Image\_A, 2: Image\_B}

**Sample output:**

Completion Code : Success

**PSU Update new workflow**

1. Copy psu binaries(.hex) files to a "Package" folder.
2. Create a manifest.txt file in Package folder. And make the entries for each vendor. As a sample, Please see the following.  
FlexPower=FPS-213-D0000293-101\_MIS-S-  
1020ADE00\_RegionB\_CS656F\_V0001.0x2917133D\_20170725.hex

Artesyn=FPS-213-D0000293-101\_MIS-S-1020ADE00\_RegionB\_CS4DE8\_V0001.0x2917133D\_20170725.hex

3. The final package folder structure looks like the following.



4. Archive the package folder as “.zip”.
5. Copy .zip folder in rack manager (set manager tftp get -s {} -f {package.zip}).
6. Execute psu update command.

“set system psu update -i {Id} -t {1-2} -f package.zip

Note: Make sure package hold appropriate Region’s binary, as an example, If system needs Region\_B image, then pack only have Region B psu images(all vendors).

## 7.50 Set system bios update

### Description:

The firmware update for the server BIOS firmware

### Syntax:

Wcscli# set system bios update [-h] -i server-id -f <file-path>

### Sample usage:

usage: set system psu update [-h] -i server-id -f </tftp/psufw.txt>

-i <Server-id> Serever id <1 to 48 ids>

-f file-path firmware file path to update.

### Sample output:

Completion Code : Success

## 7.51 Set system bmc update

### Description:

The firmware update for the server BMC firmware

### Syntax:

Wcscli# set system bmc update [-h] -i server-id -f <file-path>

### Sample usage:

---

```
usage: set system bmc update [-h] -i server-id -f </tftp/bmcfw.txt>
```

```
-i <Server-id> Serever id <1 to 48 ids>
```

```
-f file-path firmware file path to update.
```

**Sample output:**

```
Completion Code : Success
```

---

## 7.52 Set system cpld update

**Description:**

The firmware update for the server CPLD firmware

**Syntax:**

```
Wcscli# set system cpld update [-h] -i server-id -f <file-path>
```

**Sample usage:**

```
usage: set system cpld update [-h] -i server-id -f </tftp/cpldfw.txt>
```

```
-i <Server-id> Serever id <1 to 48 ids>
```

```
-f file-path firmware file path to update.
```

**Sample output:**

```
Completion Code : Success
```

---

## 7.53 Set FPGA Bypass Enable

**Description:**

This command enables bypass mode on a FPGA for a server.

**Syntax:**

```
wcscli# set system fpga bypass enable <-i server_index > [-h]
```

```
-i – Server index, the number of servers (1-48)
```

```
-h – Help, display the correct syntax
```

**Sample usage:**

To enable FPGA bypass mode for server 1, use the following command:

```
wcscli# set system fpga bypass enable -i 1
```

**Sample output:**

---



---

Completion Code: Success

---

### 7.54 Set FPGA Bypass Disable

**Description:**

This command disables bypass mode on a FPGA for a server.

**Syntax:**

wcscli# set system fpga bypass disable <-i server\_index > [-h]

-i – Server index, the number of servers (1-48)

-h – Help, display the correct syntax

**Sample usage:**

To disable FPGA bypass mode for server 1, use the following command:

wcscli# set system fpga bypass disable -i 1

**Sample output:**

Completion Code: Success

---

### 7.55 Set Tpm Physical Presence

**Description:**

This command sets TPM physical presence

**Syntax:**

wcscli# set system tpm presence [-h] -i <server-id> -p presence

server\_index - the target server number. Typically 1-48

-h - help; display the correct syntax

**Sample usage:**

wcscli# set system tpm presence -i 27

**Sample output:**

Completion Code: Success

---

### 7.56 Set System Rmedia Mount

**Description:**

This command mounts an iso image as a boot drive for the specified blade/server

---

**Note:**

- The iso image file name specified in the command below is expected to be in the TFTP location before this command is expected. Use 'set manager tftp get' command to place the file before using this command.
- After the execution of this command, setnextboot command on the blade/server should be used if this remote media is expected to be used as a boot medium.

**Syntax:**

usage: set system rmedia mount [-h] -i server-id -n rmedia-image-name

## Optional arguments:

- h, --help show this help message and exit
- b DEVICEID device id (0:Host OS or 1:Soc)
- m {1,2} image mount file type (1:hd(default) 2:cd)

## Required arguments:

- i SERVERID server-id
- n remotedriver-image-name( remote media image name)

**Sample usage:**

```
wcscli# set system rmedia mount -i 23 -n winpe.img
```

**Sample output:**

Completion Code: Success

---

## 7.57 Set System Rmedia Unmount

**Description:**

This command unmounts an iso image from the specified blade/server

**Note:**

- This command is expected to be used after the rmedia mount command has been successfully used.

**Syntax:**

usage: set system rmedia unmount [-h] -i server-id

## Optional arguments:

---

-h, --help show this help message and exit

-b DEVICEID device id (0:Host OS or 1:Soc)

-m {1,2} image mount file type (1:hd(default) 2:cd)

Required arguments:

-i SERVERID server-id

**Sample usage:**

wcscli# set system rmedia unmount -i 23

**Sample output:**

Completion Code: Success

---

### 7.58 Set System file delete

**Description:**

This command deletes the specified file from the server

**Syntax:**

set system file delete [-h] -i server-id -f file-name

optional arguments:

-h, --help show this help message and exit

required arguments:

-i server-id server-id

-f file-name file-name

**Usage:**

**Wcscli# set system file delete -i 1 -f file-name**

**Sample Output:**

Completion Code : Success

---

### 7.59 Set System file get

**Description:**

This command gets the specified file from the server (to Rack Manager TFTP location)

**Syntax:**

set system file get [-h] -i server-id -f file-name

optional arguments:

-h, --help show this help message and exit

---

---

required arguments:

- i server-id server-id
- f file-name file-name

**Usage:**

**Wcscli# set system file get -i 1 -f file-name**

**Sample Output:**

Completion Code : Success

---

## 7.60 Set System file put

**Description:**

This command puts/copies the specified file to the server (from Rack Manager TFTP location)

**Syntax:**

set system file put [-h] -i server-id -f file-name

optional arguments:

- h, --help show this help message and exit

required arguments:

- i server-id server-id
- f file-name file-name

**Usage:**

**Wcscli# set system file put -i 1 -f file-name**

**Sample Output:**

Completion Code : Success

---

## 7.61 Start Serial Session

**Description:**

This command is used to start serial session to a server. The command will open a Serial-Client-terminal for the serial session.

Users might want to open a serial session to a server for debugging purposes, to view blade boot messages, or for executing BIOS commands. A VT100 console will be provided that will continuously poll the blade for any serial session data. Any user command entered using the VT100 console will be sent to the server.

**Syntax:**

**Wcscli# start serial session <-i server\_id> [-h]**

- i - SERVERID server-id (1 to 48)
- b - DEVICEID device id (0 or 1)
- f - Force start

-b (device type) it is optional, by default it accepts 0: Host OS and 1 redirects to Soc

---

---

-f force start serial connection.

**Sample usage:**

usage: start serial session -i server-id

provide 'superuser' password when prompted

**Sample output:**

SAC>

Key board short cut to exit from SAC prompt : ~&

---

## 7.62 Stop Serial Session

**Description:**

This command is used to force the termination of an active serial session to a blade.

**Syntax:**

```
Wcscli# stop serial session <-i server_index> [-h]
```

-i - SERVERID server-id (1 to 48)

-b - DEVICEID device id (0 or 1)

-h - help; display the correct syntax

-b (device type) it is optional, by default it accepts 0: Host OS and 1 redirects to Soc

Commands to escape the serial session are as follows:

Shift + ~ + &

**Sample usage:**

To stop a serial session to a server, use the following command:

```
wcscli# stop serial session -i server_index
```

**Sample Output:**

---

## 7.63 Start JTAG service

**Description:**

Start FPGA JTAG service.

**Syntax:**

```
Wcscli# start jtag service [-h] -i <server-id>
```

-h - help; display the correct syntax

-i - the number of the server. Typically 1-48

**Sample usage:**

---

---

usage: start jtag service

#### 7.64 Stop JTAG service

##### **Description:**

Stop FPGA JTAG service.

##### **Syntax:**

Wcscli# stop jtag service [-h] -i <server-id>  
-h - help; display the correct syntax  
-i - the number of the server. Typically 1-48

##### **Sample usage:**

usage: stop jtag service

---

## 8 Switch Commands

- [Show Switch Info](#)
- [show Switch Port](#)
- [Set Switch Config](#)
- [Set Switch Upgrade](#)
- [Set Switch Reset](#)
- [Start Auxiliary Console](#)

---

### 8.1 Show Switch Info

**Description:**

Show the overall switch status.

**Syntax:**

```
Wcscli# show switch info [-h] [-i <ip>]  
-i - The IP address of the switch (optional).  
-h - help; display the correct syntax
```

**Sample usage:**

usage: show switch info

**Sample output:**

---

### 8.2 Show Switch Port

**Description:**

Show the status for a single port

**Syntax:**

```
wcscli# show switch port [-h] [-i <ip>] -p <port_num>  
-p -> The port number to query.  
-i -> The IP address of the switch (optional).  
-h - help; display the correct syntax
```

**Sample usage:**

usage: show switch port -p 1 [ip-address]

**Sample output:**

---

### 8.3 Set Switch Config

**Description:**

Update the configuration of the switch

**Syntax:**

```
wcscli# set switch config [-h] [-d <device>] [-v] [-f <path>]
```

-f -> The local path of the configuration text file to apply (optional). If this is not specified, the default configuration will be applied to the switch.

-d -> The serial device to use to communicate with the switch (optional).

-v -> Enable verbose mode. Echo the console output while applying the configuration.

-h - help; display the correct syntax

**Sample usage:**

```
Wcscli# set switch config [-d <device>] [-v] -f <path>
```

**Sample output:**

---

### 8.4 Set Switch Upgrade

**Description:**

Upgrade the firmware of the switch

**Syntax:**

```
wcscli# set switch upgrade [-h] [-d <device>] [-v] -f <path>
```

-f -> The path relative to the TFTP server root of the firmware image.

-d -> The serial device to use to communicate with the switch (optional).

-v -> Enable verbose mode. Echo the console output while upgrading the firmware.

-h - help; display the correct syntax

**Sample usage:**

```
wcscli# set switch upgrade [-h] [-d <device>] [-v] -f <path>
```

**Sample Output:**

---

### 8.5 Set Switch Reset

**Description:**

---



---

## Reset the switch

**Syntax:**

Wcscli# set switch reset [-h] [-d <device>] [-v]

-d -> The serial device to use to communicate with the switch (optional).

-v -> Enable verbose mode. Echo the console output while resetting.

-h - help; display the correct syntax

**Sample usage:**

Wcscli# set switch reset [-h] [-d <device>] [-v]

**Sample output:**

---

## 8.6 Start Auxiliary Console

**Description:**

Interactive access to the switch console

**Syntax:**

Wcscli# start auxiliary console [-h] [-d <device>] [-b <baud>] [-x]

-d -> The serial device to use to communicate with the switch (optional).

-b -> The baud rate of the serial port (optional). Defaults to 115200.

-x -> Pass Ctrl-C and Ctrl-D signals to the remote console instead terminating the console session.

Ctrl-X will be the only way to terminate the session.

-h - help; display the correct syntax

**Sample usage:**

Wcscli# start aux console [-h] [-d <device>]

**Sample output:**

---

## 9 UPS Commands

- [Show ups voltage](#)
- [Show ups power](#)
- [Show ups current](#)
- [Show ups state](#)
- [Show ups test](#)
- [Show ups config](#)
- [Show ups timer](#)
- [Show ups info](#)
- [Show ups alarm](#)

### 9.1 Show ups voltage

**Description:**

Show UPS system voltages.

**Syntax:**

```
Wcscli# show ups voltage [-h]  
-h - help; display the correct syntax
```

**Sample usage:**

usage: show ups voltage

### 9.2 Show ups power

**Description:**

Show UPS power output information

**Syntax:**

```
wcscli# show ups power [-h]  
-h - help; display the correct syntax
```

**Sample usage:**

usage: show ups power

### 9.3 Show ups current

**Description:**

Show UPS system current

**Syntax:**

```
wcscli# show ups current [-h]  
-h - help; display the correct syntax
```

**Sample usage:**

```
Wcscli# show ups current
```

---

#### 9.4 Show ups state

**Description:**

Show UPS system state

**Syntax:**

```
wcscli# show ups state [-h]  
-h - help; display the correct syntax
```

**Sample usage:**

```
Wcscli# show ups state
```

---

#### 9.5 Show ups test

**Description:**

Show UPS test results

**Syntax:**

```
Wcscli# show ups test [-h]  
-h - help; display the correct syntax
```

**Sample usage:**

```
Wcscli# show ups test
```

---

#### 9.6 Show ups config

**Description:**

Show UPS configuration

**Syntax:**

```
Wcscli# show ups config [-h]  
-h - help; display the correct syntax
```

**Sample usage:**

```
Wcscli# show ups config
```

---

## 9.7 Show ups timer

**Description:**

Show UPS startup and shutdown timers

**Syntax:**

```
Wcscli# show ups timer [-h]  
-h - help; display the correct syntax
```

**Sample usage:**

```
Wcscli# show ups timer
```

---

## 9.8 Show ups info

**Description:**

Show UPS device identifiers

**Syntax:**

```
Wcscli# show ups info [-h]  
-h - help; display the correct syntax
```

**Sample usage:**

```
Wcscli# show ups info
```

---

## 9.9 Show ups alarm

**Description:**

Show UPS alarm statuses

**Syntax:**

```
Wcscli# show ups alarm [-h]  
-h - help; display the correct syntax
```

---

**Sample usage:**

Wcscli# show ups alarm

## 10 Help Commands

### 10.1 Help

**Description:**

Shows list of categories

**Syntax:**

Wcscli# help

**Sample output:**

**wcscli# help**

Command Types (type help <topic>)

=====

show sh set start stop exit help

Command Groups (type help <topic>):

=====

version user network system manager mte switch

### 10.2 help show

Wcscli# help show

Execute ocscli show commands

usage:

show group command <parameters>

SW: RM CLI

---

### Supported groups

---

1. network
2. manager
3. system
4. user
5. switch
6. mte
7. version

## 10.3 help set

wcscli# help set

Execute ocscli set commands

usage:

set group command <parameters>

---

### Supported groups

---

1. network
2. manager
3. system
4. user
5. switch
6. **mte**

## 10.4 help start

wcscli# help start

Execute ocscli start commands.

Usage:

start command <parameters>

---

### Supported commands

---

1. serial
2. mte
3. switch

**wcscli# start mte -h**

usage: ocshell.py [-h] -u UTIL [-p [PARAMS [PARAMS ...]]] [-t TIMES]

---

**[-d DURATION]**

Execute MTE utility

optional arguments:

-h, --help        show this help message and exit  
 -u UTIL           Utility name  
 -p [PARAMS [PARAMS ...]]  
                   Utility parameters  
 -t TIMES          Run times  
 -d DURATION       Run duration

**wcscli# start serial session -h**

usage: start serial session [-h] -i SERVERID

optional arguments:

-h, --help   show this help message and exit

required arguments:

-i SERVERID   server-id (1 to 48)

**wcscli# start switch -h**

usage: start switch [-h] {console} ...

Management switch start commands

positional arguments:

{console}   Command type  
 console    Access switch console

optional arguments:

-h, --help   show this help message and exit

---

## 10.5 help stop

**wcscli# help stop**

Execute ocscli stop commands.

Usage:

stop command <parameters>

-----

Supported commands

-----

1. serial
2. mte

**wcscli# stop serial -h**

---

---

```
usage: stop serial [-h] {session} ...
```

```
Rack manager: server management start serial session command.
```

```
positional arguments:
```

```
  {session}    Program mode serial
  session      Stops serial session
```

```
optional arguments:
```

```
  -h, --help  show this help message and exit
```

---

## 10.6 help user

```
wcscli# help user
```

```
Ocscli User "show" Commands :
```

```
=====
```

```
display
```

```
Ocscli User "set" Commands:
```

```
=====
```

```
add update delete
```

```
wcscli# help system
```

```
wcscli# help network
```

```
wcscli# help manager
```

---

## wcscli# help ledon

```
ledon: This command turns the server UID LED on.
```

```
=====
```

```
Usage:
```

```
  set system led on -i {serverid}
```

```
-i -- serverid, the target server number. Typically 1-48
```

```
[-h] -help; display the correct syntax
```

```
Wcscli# sh system -h
```

```
Wcscli# set system -h
```

```
Wcscli# sh manager -h
```

```
Wcscli# set manager -h
```

```
Wcscli# set network -h
```

```
wcscli# sh system state -h
```

```
etc..
```

---



# 11 MTE Commands

## 1. Execute MTE utilities

Syntax:

```
Ocscli: start mte [-h] -u UTIL [-p [PARAMS [PARAMS ...]]] [-t TIMES] [-d DURATION]
```

**Note1:** ocs-fru, ocs-mac and ocs-adccalibration utilities expect a text file as user input. These text files are expected to be placed under Rack Manager TFTP location (via tftp get command). For this version, the filenames are hard coded and is expected as follows,

```
ocs-fru: fruupdate.txt
ocs-mac: macupdate.txt
ocs-adccalibration: adccalibration.txt
```

**Note2:** Before running network test commands (like iperf, ping etc) disable firewall using the mte firewall command (see below). You may want to enable firewall after completing the network tests.

**Note3:** After completing the mte test, as a final step, the mte lock write command (see below) shall be executed which will fuse/lock mte commands from future usage on that RM.

Available utilities:

Utility	Usage
ocs-gpio	start mte -u ocs-gpio -p setupgpio start mte -u ocs-gpio -p cleanupgpio start mte -u ocs-gpio -p portbuffer <on/off/state> start mte -u ocs-gpio -p portpresent <ID> start mte -u ocs-gpio -p portcontrol <ID> <on/off> start mte -u ocs-gpio -p portstate <ID> start mte -u ocs-gpio -p lrselect <on/off/state> start mte -u ocs-gpio -p throttlebypass <on/off/state> <id> id = 0 for rack manager, id =1 for row manager start mte -u ocs-gpio -p throttlecontrol <on/off/state> <id> id = 0 for rack manager, id =1 for row manager start mte -u ocs-gpio -p relaycontrol <ID> <on/off/state> start mte -u ocs-gpio -p debugled <ID> <on/off/state> start mte -u ocs-gpio -p getmode
ocs-pru	start mte -u ocs-pru -p GetPruVersion start mte -u ocs-pru -p clearmaxpower start mte -u ocs-pru -p getmaxpower start mte -u ocs-pru -p clearphasestatus <FEED#> start mte -u ocs-pru -p getphasestatus <FEED#> start mte -u ocs-pru -p getphasecurrent <FEED#> start mte -u ocs-pru -p getphasevoltage <FEED#> start mte -u ocs-pru -p getphasepower <FEED#>

SW: RM CLI

	start mte -u ocs-pru -p getgain <PHASE#> <0/1> start mte -u ocs-pru -p setgain <PHASE#> <0/1> <value> start mte -u ocs-pru -p getoffset <PHASE#> <0/1> start mte -u ocs-pru -p setoffset <PHASE#> <0/1> <value> start mte -u ocs-pru -p setthrottleactive <0/1> start mte -u ocs-pru -p getthrottlestatus start mte -u ocs-pru -p setthrottleenable <0/1> <0/1> first parameter is for rm throttle, second is for dc throttle start mte -u ocs-pru -p getpower start mte -u ocs-pru -p setthrottlelimit <value> start mte -u ocs-pru -p getthrottlelimit start mte -u ocs-pru -p \-v
ocs-hsc	start mte -u ocs-hsc -p GetPower start mte -u ocs-hsc -p GetInputVoltage start mte -u ocs-hsc -p GetStatus start mte -u ocs-hsc -p \-v
ocs-hdc	start mte -u ocs-hdc -p GetTemperature start mte -u ocs-hdc -p GetHumidity start mte -u ocs-hdc -p Initialize start mte -u ocs-hdc -p \-v
ocs-fru	start mte -u ocs-fru -p read <i2c bus> <i2c slave address> start mte -u ocs-fru -p write <i2c bus> <i2c slave address>
ocs-pmic	start mte -u ocs-pmic -p GetStatus start mte -u ocs-pmic -p SetFseal start mte -u ocs-pmic -p Initialize start mte -u ocs-pmic -p \-v
ocs-adccalibration	start mte -u ocs-adccalibration -p read start mte -u ocs-adccalibration -p write start mte -u ocs-adccalibration -p calibration
ocs-mac	start mte -u ocs-mac -p read start mte -u ocs-mac -p write
stream	start mte -u stream [-t TIMES] [-d DURATION]
sysbench	start mte -u sysbench [-t TIMES] [-d DURATION]
memtester	start mte -u memtester -p <mem> <loop>
ttyS0	start mte -u ttyS0
ttyS1	start mte -u ttyS1
ttyS2	start mte -u ttyS2
fio	start mte -u fio
lmbench3	start mte -u lmbench -p <seconds>
iperf	start mte -u iperf -p <tcp/udp> <IP address> start mte -u iperf -p server
minicom	start mte -u minicom
reboot	start mte -u reboot
date	start mte -u date -p <date>
hwclock	start mte -u hwclock
i2cdetect	start mte -u i2cdetect -p <i2c bus>
i2cdump	start mte -u i2cdump -p <i2c bus> <chip address>

i2cset	start mte -u i2cset -p <i2c bus> <chip address> <data address> <value>
i2cget	start mte -u i2cget -p <i2c bus> <chip address> <data address>
cpuinfo	start mte -u cpuinfo
meminfo	start mte -u meminfo
qspiinfo	start mte -u qspiinfo
mmcinfo	start mte -u mmcinfo
ping	start mte -u ping -p <IP address>
firewall	start mte -u firewall -p <on/off>
lock	start mte -u lock -p <read/write>
tfb	start mte -u tfb -p <getprov/setprov>

## 2. Show all executing MTE utilities

Syntax:

Ocscli: show mte

## 3. Stop all executing MTE utilities

Syntax:

Ocscli: stop mte

## 12 Appendix: Commonly Used of Acronyms

This section provides definitions of acronyms used in the WCS system specifications.

<b>ACPI</b> – advanced configuration and power interface	<b>DSR</b> – data set ready
<b>AHCI</b> – advanced host controller interface	<b>DTR</b> – data terminal ready
<b>AHJ</b> – authority having jurisdiction	<b>ECC</b> – error-correcting code
<b>ANSI</b> – American National Standards Institute	<b>EEPROM</b> - electrically erasable programmable read-only memory
<b>API</b> – application programming interface	<b>EIA</b> – Electronic Industries Alliance
<b>ASHRAE</b> – American Society of Heating, Refrigerating and Air Conditioning Engineers	<b>EMC</b> – electromagnetic compatibility
<b>ASIC</b> – application-specific integrated circuit	<b>EMI</b> – electromagnetic interference
<b>BCD</b> – binary-coded decimal	<b>FRU</b> – field replaceable unit
<b>BIOS</b> – basic input/output system	<b>FTP</b> – file transfer protocol
<b>BMC</b> – baseboard management controller	<b>GPIO</b> – general purpose input output
<b>CFM</b> – cubic feet per minute (measure of volume flow rate)	<b>GUID</b> – globally unique identifier
<b>CM</b> – Chassis Manager	<b>HBI</b> – high business intelligence
<b>CMOS</b> – complementary metal–oxide–semiconductor	<b>HCK</b> – Windows Hardware Certification Kit
<b>COLO</b> – co-location	<b>HMD</b> – hardware monitoring device
<b>CTS</b> – clear to send	<b>HT</b> – hyperthreading
<b>DCMI</b> – Data Center Manageability Interface	<b>I<sup>2</sup>C</b> – inter-integrated circuit
<b>DDR3</b> – double data rate type 3	<b>IBC</b> – international building code
<b>DHCP</b> – dynamic host configuration protocol	<b>IDE</b> – integrated development environment
<b>DIMM</b> – dual inline memory module	<b>IEC</b> - International Electrotechnical Commission
<b>DPC</b> - DIMMs per memory channel	<b>IOC</b> – I/O controller
<b>DRAM</b> – Dynamic Random Access Memory	<b>IPMI</b> – intelligent platform management interface
	<b>IPsec</b> – IP security

<b>ITPAC</b> – IT pre-assembled components	<b>Ph-ph</b> – phase to phase
<b>JBOD</b> – “just a bunch of disks”	<b>Ph-N</b> – phase to neutral
<b>KCS</b> – keyboard controller style	<b>PNP</b> – plug and play
<b>L2</b> – layer 2	<b>POST</b> – power-on self-test
<b>LAN</b> – local area network	<b>PSU</b> – power supply unit
<b>LFF</b> – large form factor	<b>PWM</b> – pulse-width modulation
<b>LPC</b> – low pin count	<b>PXE</b> – preboot execution environment
<b>LS</b> – least significant	<b>QDR</b> – quad data rate
<b>LUN</b> – logical unit number	<b>QFN</b> – quad flat package no-lead
<b>MAC</b> – media access control	<b>QPI</b> – Intel QuickPath Interconnect
<b>MDC</b> – modular data center containers	<b>QSFP</b> – Quad small form-factor pluggable
<b>MLC</b> – multi-level cell	<b>RAID</b> – redundant array of independent disks
<b>MTBF</b> – mean time between failures	<b>REST</b> – representational state transfer
<b>MUX</b> – multiplexer	<b>RM</b> – Rack Manager
<b>NIC</b> – network interface card	<b>RMA</b> – remote management agent
<b>NUMA</b> – non-uniform memory access	<b>ROC</b> – RAID-on-chip controller
<b>OOB</b> – out of band	<b>RSS</b> – receive-side scaling
<b>OSHA</b> – Occupational Safety & Health Administration	<b>RTS</b> – request to send
<b>OTS</b> – off the shelf	<b>RU</b> – rack unit
<b>PCB</b> – printed circuit board	<b>RxD</b> – received data
<b>PCIe</b> – peripheral component interconnect express	<b>SAS</b> – serial-attached small computer system interface (SCSI)
<b>PCH</b> – platform control hub	<b>SATA</b> – serial AT attachment
<b>PDB</b> – power distribution backplane	<b>SCK</b> – serial clock
<b>PDU</b> – power distribution unit	<b>SCSI</b> – small computer system interface
	<b>SDA</b> – serial data signal

<b>SDR</b> – sensor data record	<b>TOR</b> – top of rack
<b>SFF</b> – small form factor	<b>TPM</b> – trusted platform module
<b>SFP</b> - small form-factor pluggable	<b>TxD</b> – transmit data
<b>SMBUS</b> – systems management bus	<b>U</b> – rack unit
<b>SMBIOS</b> – systems management BIOS	<b>UART</b> – universal asynchronous receiver/transmitter
<b>SOL</b> – serial over LAN	<b>UEFI</b> – unified extensible firmware interface
<b>SPI</b> – serial peripheral interface	<b>UL</b> – Underwriters Laboratories
<b>SSD</b> – solid-state drive	<b>UPS</b> – uninterrupted power supply
<b>TB</b> – tray backplane	<b>Vpp</b> – voltage peak to peak
<b>TDP</b> – thermal design power	<b>WMI</b> –Windows Management Interface
<b>TM</b> – tray midplane	

SW: RM CLI