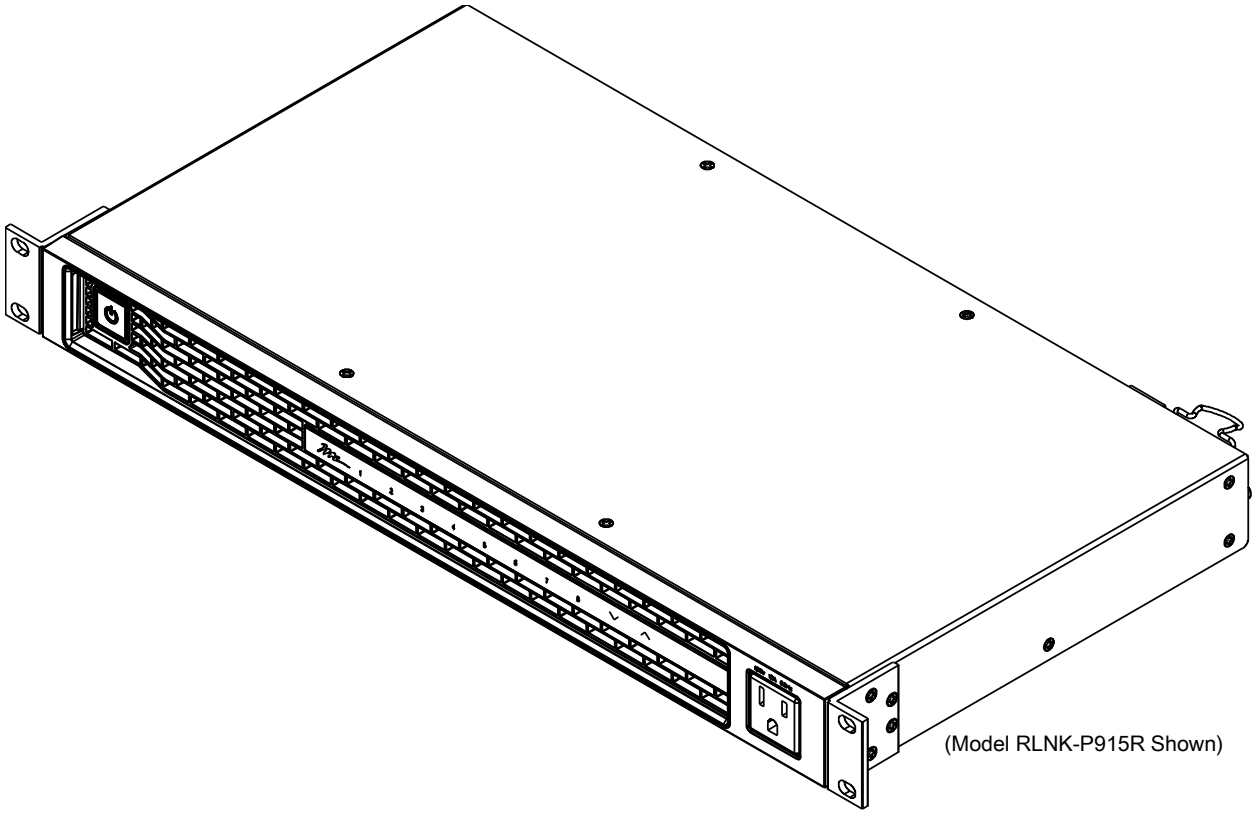
Advanced User Manual

Premium+ PDU with RackLink™

Monitor | Control | Alert | Report | Analyze



(Model RLNK-P915R Shown)

THANK YOU

Thank you for assisting our customers with their Premium+ PDU with RackLink™ product. Please read these instructions thoroughly before installing or assembling this product.

Middle Atlantic Products

I-00852 Rev B

Contents

Chapter 1: Introduction

Thank you for purchasing Middle Atlantic Products Premium+ PDU with RackLink (referred to in this document as PDU). This Advanced User Manual provides explanations for more advanced functionality when using your PDU.

The complete set of instructions for your PDU is available from www.middleatlantic.com/resources/power-downloads.aspx and includes the following documents:

- The Quick Start Guide (I-00827 for Rackmount Units, I-00864 for Compact Units)
- The User Manual (I-00826)
- The Advanced User Manual (I-00852)
- The Environmental Sensors User Manual (I-00853)

Chapter 2: Configuring the PDU

You can initially configure the PDU by connecting it to a computer, or to a TCP/IP network that supports DHCP.

Configuration over a DHCP-enabled network:

1. Connect the PDU to a DHCP IPv4 network. Refer to ***Connecting the PDU to Your Network*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.
2. Retrieve the DHCP-assigned IPv4 address. Use the front panel display to retrieve it. See ***Device Information*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.
3. Launch a web browser to configure the PDU. See ***Login*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx..

Configuration using a connected computer:

1. Connect the PDU to a computer. See ***Connecting the PDU to a Computer*** (on page 7).
2. Use the connected computer to configure the PDU via the command line or web interface.
   - Command line interface: See ***Initial Network Configuration via CLI*** (on page 9).
   - Web interface: Launch the web browser on the computer, and type the link-local IP address or *pdu.local* to access the PDU. Refer to ***Log In, Log Out, and Password Change*** in the Premium+ PDU with RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.

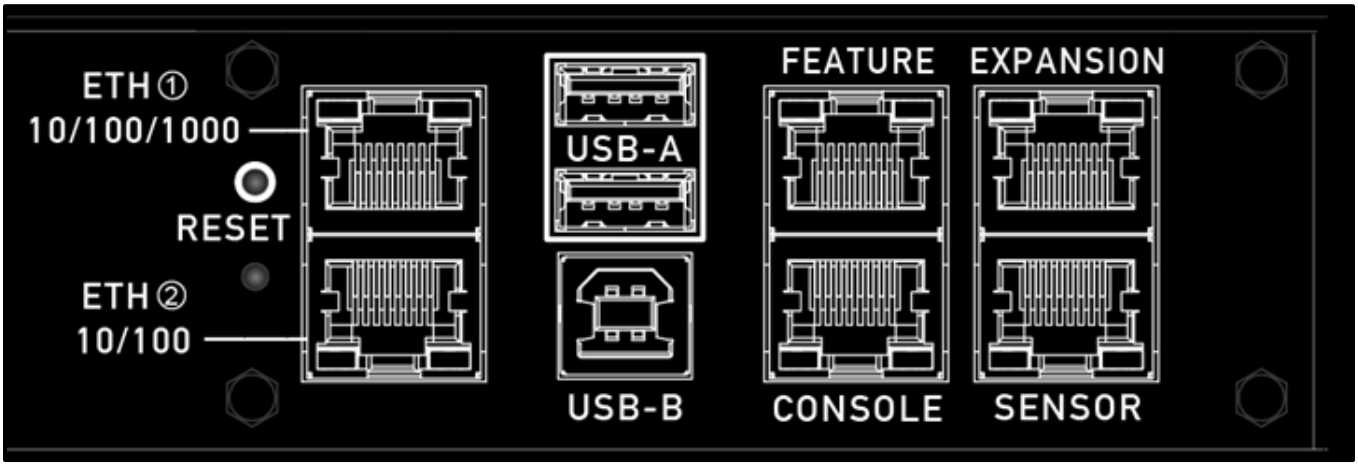
For link-local IP address retrieval, see ***Device Information*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.

Tip: To configure a number of PDUs quickly, see ***Chapter 5: Bulk Configuration Methods*** *(on page 134).*

Connecting the PDU to a Computer

The PDU can be connected to a computer for configuration via one of the following ports.

- USB-B port (male)
- ETHERNET port (female)
- CONSOLE port



To use the command line interface (CLI) for configuration, establish an CONSOLE or USB connection.

To use a web browser for configuration, make a network connection to the computer. The PDU is automatically configured with the following link-local addressing in any network without DHCP available:

- *https://169.254.x.x* (where x is a number)
- *https://pdu.local*

See ***APIPA and Link-Local Addressing*** (on page 152).

Establish one of the following connections to a computer.

Serial connection for CONSOLE connector on the PDU:

1. Obtain a third-party RJ-45 to female DB9 adapter/cable, such as a Cisco console adapter/cable.
2. Connect the RJ-45 end of the cable to the CONSOLE port on the PDU.
3. Connect the other end to your computer's RS-232 port (COM).
4. Perform ***Initial Network Configuration via CLI*** (on page 9).

USB connection:

1. A USB-to-serial driver is required in Windows®. Install this driver before connecting the USB cable. See ***Installing the USB-to-Serial Driver (Optional)*** (on page 8).
2. Connect a USB cable between the PDU's USB-B port and a computer's USB-A port.
3. Perform ***Initial Network Configuration via CLI*** (on page 9).

Note: Not all serial-to-USB converters work properly with the PDU, so it is not recommended for use.

Direct network connection:

The Ethernet port must be enabled for this connection to work properly. Per default, the Ethernet port is enabled.

1. Connect one end of a standard network patch cable (Cat5e or better, not provided) to the ETHERNET port of the PDU.
2. Connect the other end to a computer's Ethernet port.
3. On the connected computer, launch a web browser to access the PDU, using either link-local addressing: *pdu.local* or *192.168.1.200.* For more information, refer to ***Log in, Log Out, and Password Change*** in the Premium+ Sereis PDU with RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.

Installing the USB-to-Serial Driver (Optional)

The PDU can emulate a USB-to-serial converter over a USB connection. A USB-to-serial driver named "RackLink Serial Console" is required for Microsoft® Windows® operating systems.

Download the Windows driver for USB serial console at www.middleatlantic.com/resources/power-downloads.aspx. The downloaded driver's name is *racklink-serial-setup-<n>.exe*, where <n> represents the file's version number.

There are two ways to install this driver: automatic and manual installation. Automatic driver installation is highly recommended.

Automatic driver installation in Windows®:

1. Make sure the PDU is NOT connected to the computer via a USB cable.
2. Run racklink-serial-setup-<n>.exe on the computer and follow online instructions to install the driver.

Note: If any Windows security warning appears, accept it to continue the installation.

3. Connect the PDU to the computer via a USB cable. The driver is automatically installed.

Manual driver installation in Windows®:

1. Make sure the PDU has been connected to the computer via a USB cable.

2. The computer detects the new device and the "Found New Hardware Wizard" dialog appears.
   - If this dialog does not appear, choose Control Panel > System > Hardware > Device Manager, right-click the *RackLink Serial Console*, and choose Update Driver.
3. Select the option of driver installation from a specific location, and then specify the location where both *dominion-serial.inf* and *dominion-serial.cat* are stored.

   Note: If any Windows security warning appears, accept it to continue the installation.
4. Wait until the installation is complete.

Note: If the PDU enters the disaster recovery mode when the USB serial driver is not installed yet, it may be shown as a 'GPS camera' in the Device Manager on the computer connected to it.

In Linux:

No additional drivers are required, but you must provide the name of the tty device, which can be found in the output of the "dmesg" after connecting the PDU to the computer. Usually the tty device is "/dev/ttyACM#" or "/dev/ttyUSB#," where # is an integer number.

For example, if you are using the kermit terminal program, and the tty device is "/dev/ttyACM0," perform the following commands:

> set line /dev/ttyACM0

> Connect

Initial Network Configuration via CLI

After the PDU is connected to your network, you must provide it with an IP address and some additional networking information.

This section describes the initial network configuration via a serial CONSOLE or USB connection. To configure the network settings using the web interface, see ***Configuring Network Settings*** in the Premium+ Sereis PDU with RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.

To configure the PDU:

1. On the computer connected to the PDU, open a communications program such as HyperTerminal or PuTTY.
2. Select the appropriate COM port, and set the following port settings:
   - Bits per second = 115200 (115.2Kbps)
   - Data bits = 8

- Stop bits = 1
- Parity = None
- Flow control = None

Tip: For a USB connection, you can determine the COM port by choosing Control Panel > System > Hardware > Device Manager, and locating the "RackLink Serial Console" under the Ports group.

3. In the communications program, press Enter to send a carriage return to the PDU.
4. The PDU prompts you to log in. Both user name and password are case sensitive.
   a. Username: `admin`
   b. Password: `admin` (or a new password if you have changed it).
5. If prompted to change the default password, change or ignore it.
   - To change it, follow onscreen instructions to type your new password.
   - To ignore it, simply press Enter.

   The # prompt appears.
6. Type `config` and press Enter.
7. To configure network settings, type appropriate commands and press Enter. Refer to the following commands list. CLI commands are case sensitive.
8. After finishing the network settings, type `apply` to save changes. To abort, type `cancel`.

Commands for wired networking:

The <ipvX> variable in the following commands is either *ipv4* or *ipv6*, depending on the type of IP protocol you are configuring.

For the PDU, replace the variable <ETH> with either 'ETH1' or 'ETH2', depending on which Ethernet port you are configuring.

- **General IP settings:**

To set or enable	Use this command
IPv4 or IPv6 protocol	`network <ipvX> interface <ETH> enabled <option>` <option> = *true*, or *false*
IPv4 configuration method	`network ipv4 interface <ETH> configMethod <mode>` <mode> = *dhcp* (default) or *static*

To set or enable	Use this command
IPv6 configuration method	`network ipv6 interface <ETH> configMethod <mode>` <mode> = *automatic* (default) or *static*
Preferred host name (optional)	`network <ipvX> interface <ETH> preferredHostName <name>` <name> = preferred host name
IP address returned by the DNS server	`network dns resolverPreference <resolver>` <resolver> = *preferV4* or *preferV6*

- **Static IP configuration:**

To set	Use this command
Static IPv4 or IPv6 address	`network <ipvX> interface <ETH> address <ip address>` <ip address> = static IP address, with a syntax similar to the example below. ▪ Example: *192.168.7.9/24*
Static IPv4 or IPv6 gateway	`network <ipvX> gateway <ip address>` <ip address> = gateway's IP address
IPv4 or IPv6 primary DNS server	`network dns firstServer <ip address>` <ip address> = DNS server's IP address
IPv4 or IPv6 secondary DNS server	`network dns secondServer <ip address>` <ip address> = DNS server's IP address
IPv4 or IPv6 third DNS server	`network dns thirdServer <ip address>` <ip address> = DNS server's IP address

Commands for wireless networking:

- **General wireless settings**:

To set or enable	Use this command
Wireless interface	`network wireless enabled <option>` <option> = *true*, or *false*
SSID	`network wireless SSID <ssid>` <ssid> = SSID string

To set or enable	Use this command
BSSID	`network wireless BSSID <bssid>` <bssid> = AP MAC address or *none*
802.11n protocol	`network wireless enableHT <option>` <option> = *true*, or *false*
Authentication method	`network wireless authMethod <method>` <method> = *psk* or *eap*
PSK	`network wireless PSK <psk>` <psk> = PSK string
EAP outer authentication	`network wireless eapOuterAuthentication <outer_auth>` <outer_auth> = *PEAP*
EAP inner authentication	`network wireless eapInnerAuthentication <inner_auth>` <inner_auth> = *MSCHAPv2*
EAP identity	`network wireless eapIdentity <identity>` <identity> = your user name for EAP authentication
EAP password	`network wireless eapPassword` When prompted to enter the password for EAP authentication, type the password.
EAP CA certificate	`network wireless eapCACertificate` When prompted to enter the CA certificate, open the certificate with a text editor, copy and paste the content into the communications program.

The content to be copied from the CA certificate does NOT include the first line containing "BEGIN CERTIFICATE" and the final line containing "END CERTIFICATE." If a certificate is installed, configure the following:

In order to	Use this command
Verify the certificate	`network wireless enableCertVerification <option1>` <option1> = *true* or *false*
Accept an expired or not valid certificate	`network wireless allowOffTimeRangeCerts <option2>` <option2> = *true* or *false*

In order to	Use this command
Make the connection successful by ignoring the "incorrect" system time	`network wireless allowConnectionWithIncorrectClock <option3>` <option3> = *true* or *false*

- **Wireless IPv4 / IPv6 settings:**

Commands for wireless IP settings are identical to those for wired networking. Just replace the variable <ETH> with the word 'wireless'. The following illustrates a few examples.

To set or enable	Use this command
IPv4 configuration method	`network ipv4 interface WIRELESS configMethod <mode>` <mode> = *dhcp* (default) or *static*
IPv6 configuration method	`network ipv6 interface WIRELESS configMethod <mode>` <mode> = *automatic* (default) or *static*

To verify network settings:

After exiting the above configuration mode and the # prompt re-appears, type this command to verify all network settings.

```
show network
```

The IP address configured may take seconds to take effect.

Chapter 3: Additional PDU Information

Reserving IP Addresses in DHCP Servers

The PDU uses its serial number as the client identifier in the DHCP request. Therefore, to successfully reserve an IP address for the PDU in a DHCP server, use the PDU's serial number as the unique ID instead of the MAC address.

Since all network interfaces of the PDU can be simultaneously enabled and configured with diverse static IP addresses, the client identifier of each network interface is different. The main difference is the absence/presence of a suffix, which is the interface name added to the end of the serial number. The table below lists the client identifiers of all network interfaces.

Interface	Client identifier
ETHERNET (PDU)	serial number
WIRELESS	serial number plus the uppercase suffix "-WIRELESS"

You can reserve the IP addresses of more than one interfaces in the DHCP server if preferred.

Reserving IP in Windows

To reserve the IP address of any network interface in the Windows DHCP server, you must convert that interface's client identifier into *hexadecimal* ASCII codes.

For each interface's client identifier, see ***Reserving IP Addresses in DHCP Servers*** (on page 14).

In this example, it is assumed that the PDU serial number is PEG1A00003.

▶ **Windows IP address reservation example:**

1. Convert the client identifier of the desired network interface into ASCII codes (hexadecimal).

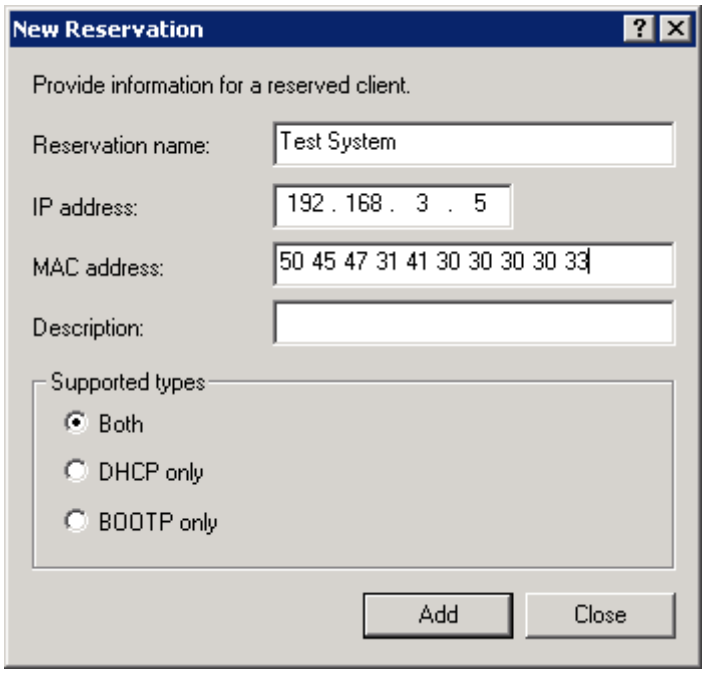
Interface	Client identifier conversion
ETHERNET (PDU)	PEG1A00003 = 50 45 47 31 41 30 30 30 30 33
WIRELESS	PEG1A00003**-WIRELESS** = 50 45 47 31 41 30 30 30 30 33 **2D 57 49 52 45 4C 45 53 53** ▪ The suffix comprising the dash symbol and the word "WIRELESS" is also converted.

2. In your DHCP server, bring up the New Reservation dialog, and separate the converted ASCII codes with spaces.

   For example, to reserve the IP address of the ETHERNET or ETH1 interface, enter the following data in the dialog.

Field	Data entered
IP address	The IP address you want to reserve.

Field	Data entered
MAC address	The following ASCII codes. `50 45 47 31 41 30 30 30 30 33`
Other fields	Configure as needed.



Reserving IP in Linux

There are two methods to reserve the IP address of any network interface in the standard Linux DHCP server (ISC DHCP server):

- Convert an interface's client identifier into *hexadecimal* ASCII codes.
- Use an interface's original client identifier without converting it into ASCII codes.

For each interface's client identifier, see ***Reserving IP Addresses in DHCP Servers*** (on page 14).

In the following examples, it is assumed that the PDU serial number is PEG1A00003, and the IP address you want to reserve is 192.168.20.1.

▶ Example with ASCII code conversion:

1. Convert the client identifier of the desired network interface into ASCII codes (*hexadecimal*).

Interface	Client identifier conversion
ETHERNET (PDU)	PEG1A00003 = 50 45 47 31 41 30 30 30 30 33

Interface	Client identifier conversion
WIRELESS	PEG1A00003**-WIRELESS** = 50 45 47 31 41 30 30 30 30 33 **2D 57 49 52 45 4C 45 53 53** ▪ The suffix comprising the dash symbol and the word "WIRELESS" is also converted.

2. Separate the converted ASCII codes with a colon, and a prefix "00:" must be added to the beginning of the converted codes.

   For example, the *converted* client identifier of the ETHERNET or ETH1 interface looks like the following:
   `00:50:45:47:31:41:30:30:30:30:33`
3. Now enter the converted client identifier with the following syntax.

```
host rlnk {
option dhcp-client-identifier = 00:50:45:47:31:41:30:30:30:30:33;
fixed-address 192.168.20.1;
}
```

▶ **Example without ASCII code conversion:**

1. Use the original client identifier of the desired network interface. DO NOT convert them into ASCII codes.
2. A prefix "\000" must be added to the beginning of the client identifier.

   For example, the client identifier of the ETHERNET or ETH1 interface looks like the following:

   `\000`PEG1A00003
3. Now enter the original client identifier with the following syntax. The client identifier is enclosed in quotation marks.

```
host rlnk {
option dhcp-client-identifier = "\000PEG1A00003";
fixed-address 192.168.20.1;
}
```

Data for BTU Calculation

To calculate the heat (BTU/hr), use the following power data according to your model type in the BTU calculation formula.

Model name	Maximum power (Watt)
PDU	10 W

Ways to Probe Existing User Profiles

This section indicates available ways to query existing user accounts on the PDU.

- With SNMP v3 activated, you get the "user unknown" error when the user name used to authenticate does not exist.
- Any user with the permission to view event rules can query all local existing users via JSON RPC.
- Any user with the permission to view the event log may get information about existing users from the log entries.
- Any authenticated users can query currently existing connection sessions, which show a list of associated user names.

Chapter 4: Using the Command Line Interface

This section explains how to use the command line interface (CLI) to administer a PDU.

CLI commands are case sensitive.

About the Interface

The PDU provides a command line interface that enables data center administrators to perform some basic management tasks.

Using this interface, you can do the following:

- Reset the PDU
- Display the PDU and network information, such as the device name, firmware version, IP address, and so on
- Configure the PDU and network settings
- Troubleshoot network problems

You can access the interface over a local connection using a terminal emulation program such as HyperTerminal, or via a Telnet or SSH client such as PuTTY.

Note: Telnet access is disabled by default because it communicates openly and is thus insecure. To enable Telnet, refer to ***Changing Telnet Settings*** *in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.*

Logging in to the CLI

Logging in via HyperTerminal over a local connection is a little different than logging in using SSH or Telnet.

If a security login agreement has been enabled, you must accept the agreement in order to complete the login. Users are authenticated first and the security banner is checked afterwards.

With HyperTerminal

You can use any terminal emulation programs for local access to the command line interface.

This section illustrates HyperTerminal, which is part of Windows operating systems prior to Windows Vista.

To log in using HyperTerminal:

1. Connect your computer to the PDU via a local connection.

2. Launch HyperTerminal on your computer and open a console window. When the window first opens, it is blank.

   Make sure the COM port settings use this configuration:

   - Bits per second = 115200 (115.2Kbps)
   - Data bits = 8
   - Stop bits = 1
   - Parity = None
   - Flow control = None

   Tip: For a USB connection, you can determine the COM port by choosing Control Panel > System > Hardware > Device Manager, and locating the "RackLink Serial Console" under the Ports group.

3. In the communications program, press Enter to send a carriage return to the PDU. The Username prompt appears.

```
Username: _
```

4. Type a name and press Enter. The name is case sensitive. Then you are prompted to enter a password.

```
Username: admin
Password: _
```

5. Type a password and press Enter. The password is case sensitive.

   After properly entering the password, the # or > system prompt appears. For more information, see ***Different CLI Modes and Prompts*** (on page 20).

   Tip: The "Last Login" information, including the date and time, is also displayed if the same user profile was used to log in to the product's web interface or CLI.

6. You are now logged in to the command line interface and can begin administering the PDU.

With SSH or Telnet

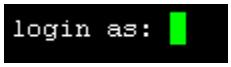
You can remotely log in to the command line interface (CLI) using an SSH or Telnet client, such as PuTTY.

Note: PuTTY is a free program you can download from the Internet. See PuTTY's documentation for details on configuration.

To log in using SSH or Telnet:

1. Ensure SSH or Telnet has been enabled. For more information, refer to ***Configuring Network Services*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.

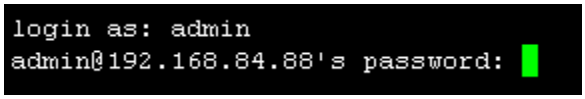
2. Launch an SSH or Telnet client and open a console window. A login prompt appears.



3. Type a name and press Enter. The name is case sensitive.

Note: If using the SSH client, the name must NOT exceed 25 characters. Otherwise, the login fails.

Then you are prompted to enter a password.



4. Type a password and press Enter. The password is case sensitive.
5. After properly entering the password, the # or > system prompt appears. For more information, see ***Different CLI Modes and Prompts*** (on page 20).

Tip: The "Last Login" information, including the date and time, is also displayed if the same user profile was used to log in to this product's web interface or CLI.

6. You are now logged in to the command line interface and can begin administering the PDU.

Different CLI Modes and Prompts

Depending on the login name you use and the mode you enter, the system prompt in the CLI varies.

- User Mode: When you log in as a normal user, who may not have full permissions to configure the PDU, the **>** prompt appears.
- Administrator Mode: When you log in as an administrator, who has full permissions to configure the PDU, the **#** prompt appears.
- Configuration Mode: You can enter the configuration mode from the administrator or user mode. In this mode, the prompt changes to **config:#** or **config:>** and you can change the PDU and network configurations. See ***Entering Configuration Mode*** (on page 42).
- Diagnostic Mode: You can enter the diagnostic mode from the administrator or user mode. In this mode, the prompt changes to **diag:#** or **diag:>** and you can perform the network troubleshooting commands, such as the ping command. See ***Entering Diagnostic Mode*** (on page 130).

Closing a Local Connection

Close the window or terminal emulation program when you finish accessing a PDU over the local connection.

When accessing or upgrading multiple PDUs, do not transfer the local connection cable from one device to another without closing the local connection window first.

Help Command

The help (?) command shows a list of main CLI commands available for the current mode. This is helpful when you are not familiar with CLI commands.

- **Help command under the administrator mode:**

  ```
  #          ?
  ```

- **Help command under the configuration mode:**

  ```
  config:#   ?
  ```

- **Help command under the diagnostic mode:**

  ```
  diag:#     ?
  ```

Press Enter after typing the help command, and a list of main commands for the current mode is displayed.

Tip: You can check what parameters are available for a specific CLI command by adding the help command to the end of the queried command. See ***Querying Available Parameters for a Command*** *(on page 21).*

Querying Available Parameters for a Command

If you are not sure what commands or parameters are available for a particular type of CLI command or its syntax, you can have the CLI show them by adding a space and the help command (?) to the end of that command. A list of available parameters and their descriptions will be displayed.

The following shows a few query examples.

- **To query available parameters for the "show" command:**

  ```
  #          show ?
  ```

- **To query available parameters for the "show user" command:**

```
#           show user ?
```

- **To query available network configuration parameters:**

```
config:#    network ?
```

- **To query available role configuration parameters:**

```
config:#    role ?
```

- **To query available parameters for the "role create" command:**

```
config:#    role create ?
```

Showing Information

You can use the show commands to view current settings or the status of the PDU or part of it, such as the IP address, networking mode, firmware version, states or readings of internal or external sensors, user profiles, and so on.

Some "show" commands have two formats: one with the parameter "details" and the other without. The difference is that the command without the parameter "details" displays a shortened version of information while the other displays in-depth information.

After typing a "show" command, press Enter to execute it.

*Note: Depending on your login name, the # prompt may be replaced by the > prompt. See **Different CLI Modes and Prompts** (on page 20).*

Network Configuration

This command shows all network configuration and all network interfaces' information, such as the IP address, MAC address, the Ethernet interface's duplex mode, and the wireless interface's status/settings.

```
#       show network
```

IP Configuration

This command shows the IP-related configuration only, such as IPv4 and IPv6 configuration, address(es), gateway, and subnet mask.

*Tip: To show IPv4-only and IPv6-only configuration data, see **IPv4-Only or IPv6-Only Configuration** (on page 23).*

```
#       show network ip common
```

To show the IP-related configuration of a specific network interface, use the following command.

```
#       show network ip interface <ETH>
```

Variables:

- <ETH> is one of the network interfaces: *ethernet (or ETH1/ETH2)*, *wireless* or *all*.

Option	Description
eth1	Show the IP-related configuration of the ETH1 interface.
eth2	Show the IP-related configuration of the ETH2 interface.
wireless	Show the IP-related configuration of the WIRELESS interface.
all	Show the IP-related configuration of all interfaces. You can type the CLI command without the word 'all.' For example, *show network ip interface*.

IPv4-Only or IPv6-Only Configuration

To show IPv4-only configuration or IPv6-only configuration, use any of the following commands.

Tip: To show both IPv4 and IPv6 configuration data, see ***IP Configuration*** *(on page 22).*

- **To show all IPv4 configuration:**

  ```
  #       show network ipv4 common
  ```

- **To show all IPv6 configuration:**

  ```
  #       show network ipv6 common
  ```

- **To show the IPv4 configuration of a specific network interface:**

  ```
  #       show network ipv4 interface <ETH>
  ```

- **To show the IPv6 configuration of a specific network interface:**

  ```
  #       show network ipv6 interface <ETH>
  ```

Variables:

- <ETH> is one of the network interfaces: *ethernet (or ETH1/ETH2)*, *wireless* or *all*.

Option	Description
eth1	Show the IPv4 or IPv6 configuration of the ETH1 interface.
eth2	Show the IPv4 or IPv6 configuration of the ETH2 interface.
all	Show the IPv4 or IPv6 configuration of all interfaces. You can type the CLI command without the word 'all.' For example, *show network ipv4 interface*.

Network Interface Settings

This command shows the specified network interface's information which is NOT related to IP configuration. For example, the Ethernet port's LAN interface speed and duplex mode, or the wireless interface's SSID parameter and authentication protocol.

```
#       show network interface <ETH>
```

Variables:

- <ETH> is one of the network interfaces: *ethernet (or ETH1/ETH2)*, *wireless* or *all*.

Option	Description
eth1	Show the IPv4 or IPv6 configuration of the ETH1 interface.
eth2	Show the IPv4 or IPv6 configuration of the ETH2 interface.
all	Show the non-IP settings of all interfaces. You can type the CLI command without the word 'all.' For example, *show network interface*.

Network Service Settings

This command shows the network service settings only, including the Telnet setting, TCP ports for HTTP, HTTPS, SSH and Modbus/TCP services, and SNMP settings.

```
#       show network services <option>
```

Variables:

- <option> is one of the options: *all*, *http*, *https*, *telnet*, *ssh*, *snmp*, *modbus* and *zeroconfig*.

Option	Description
all -- OR – [blank]	Displays the settings of all network services, including HTTP, HTTPS, Telnet, SSH and SNMP.
http	Only displays the TCP port for the HTTP service.
https	Only displays the TCP port for the HTTPS service.
telnet	Only displays the settings of the Telnet service.
ssh	Only displays the settings of the SSH service.
snmp	Only displays the SNMP settings.
modbus	Only displays the settings of the Modbus/TCP service.
zeroconfig	Only displays the settings of the zero configuration advertising.

PDU Configuration

This command shows the PDU configuration, such as the device name, firmware version and model type.

```
#       show pdu
```

To show detailed information, add the parameter "details" to the end of the command.

```
#       show pdu details
```

Outlet Information

This command syntax shows the outlet information.

```
#       show outlets <n>
```

To show detailed information, add the parameter "details" to the end of the command.

```
#       show outlets <n> details
```

Variables:

- <n> is one of the options: *all*, or a number.

Option	Description
all -- OR -- [blank]	Displays the information for all outlets.
A specific outlet number	Displays the information for the specified outlet only.

Displayed information:

- Without the parameter "details," only the outlet name and state are displayed.
- With the parameter "details," more outlet information is displayed in addition to the state, such as rated current, voltage, active power, active energy, and outlet settings.

Inlet Information

This command syntax shows the inlet information.

```
#       show inlets <n>
```

To show detailed information, add the parameter "details" to the end of the command.

```
#       show inlets <n> details
```

Variables:

- <n> is one of the options: *all*, or a number.

Option	Description
all -- OR -- [blank]	Displays the information for all inlets.
A specific inlet number	Displays the information for the specified inlet only. An inlet number needs to be specified only when there are more than 1 inlet on your PDU.

Displayed information:

- Without the parameter "details," only the inlet's name and RMS current are displayed.
- With the parameter "details," more inlet information is displayed in addition to the inlet name and RMS current, such as the inlet's RMS voltage, active power and active energy.

Overcurrent Protector Information

This command is only available for models with overcurrent protectors for protecting outlets.

This command syntax shows the overcurrent protector information, such as a circuit breaker or a fuse.

```
#       show ocp <n>
```

To show detailed information, add the parameter "details" to the end of the command.

```
#       show ocp <n> details
```

Variables:

- <n> is one of the options: *all*, or a number.

Option	Description
all -- OR -- [blank]	Displays the information for all overcurrent protectors.
A specific overcurrent protector number	Displays the information for the specified overcurrent protector only.

Displayed information:

- Without the parameter "details," only the overcurrent protector status and name are displayed.
- With the parameter "details," more overcurrent protector information is displayed in addition to status, such as the rating and RMS current value.

Date and Time Settings

This command shows the current date and time settings on the PDU.

```
#       show time
```

To show detailed information, add the parameter "details" to the end of the command.

```
#       show time details
```

Default Measurement Units

This command shows the default measurement units applied to the PDU web and CLI interfaces across all users, especially those users authenticated through remote authentication servers.

```
#       show user defaultPreferences
```

Note: If a user has set his/her own preferred measurement units or the administrator has changed any user's preferred units, the web and CLI interfaces show the preferred measurement units for that user instead of the default ones after that user logs in to the PDU. See ***Existing User Profiles*** *(on page 34) for the preferred measurement units for a specific user.*

Environmental Sensor Information

This command syntax shows the environmental sensor's information.

```
#       show externalsensors <n>
```

To show detailed information, add the parameter "details" to the end of the command.

```
#       show externalsensors <n> details

External sensor 3 ('Temperature 1')
Sensor type: Temperature
Reading:    31.8 deg C (normal)


Serial number:         AEI0950133
Description:           Not configured
Location:             X Not configured
                     Y Not configured
                     Z Not configured
Position:               Port 1
Using default thresholds: yes
```

Variables:

- <n> is one of the options: *all*, or a number.

Option	Description
all -- OR -- [blank]	Displays the information of all environmental sensors.
A specific environmental sensor number*	Displays the information for the specified environmental sensor only.

* The environmental sensor number is the ID number assigned to the sensor, which can be found on the Peripherals page of the PDU web interface.

Displayed information:

- Without the parameter "details," only the sensor ID, sensor type and reading are displayed.

  Note: A state sensor displays the sensor state instead of the reading.

- With the parameter "details," more information is displayed in addition to the ID number and sensor reading, such as the serial number, sensor position, and X, Y, and Z coordinates.

Note: DPX sensor packages do not provide chain position information.

Environmental Sensor Package Information

Different from the "`show externalsensors`" commands, which show the reading, status and configuration of an individual environmental sensor, the following command shows the information of all connected environmental sensor packages, each of which may contain more than one sensor or actuator.

```
#       show peripheralDevicePackages
```

Information similar to the following is displayed. An environmental sensor package is a peripheral device package.

```
Peripheral Device Package 1
Serial Number:    AEI7A00022
Package Type:     DPX-T1H1
Position:         Port 1
Package State:    operational
Firmware Version: Not available

Peripheral Device Package 2
Serial Number:    AEI7A00021
Package Type:     DPX-T3H1
Position:         Port 1
Package State:    operational
Firmware Version: Not available
```

Actuator Information

This command syntax shows an actuator's information.

```
#       show actuators <n>
```

To show detailed information, add the parameter "details" to the end of the command.

```
#       show actuators <n> details
```

Variables:

- <n> is one of the options: *all*, or a number.

Option	Description
all -- OR -- [blank]	Displays the information for all actuators.
A specific actuator number*	Displays the information for the specified actuator only.

* The actuator number is the ID number assigned to the actuator. The ID number can be found using the PDU web interface or CLI. It is an integer starting at 1.

Displayed information:

- Without the parameter "details," only the actuator ID, type and state are displayed.
- With the parameter "details," more information is displayed in addition to the ID number and actuator state, such as the serial number and X, Y, and Z coordinates.

Outlet Sensor Threshold Information

This command syntax shows the specified outlet sensor's threshold-related information.

```
#       show sensor outlet <n> <sensor type>
```

To show detailed information, add the parameter "details" to the end of the command.

```
#       show sensor outlet <n> <sensor type> details
```

Variables:

- <n> is the number of the outlet whose sensors you want to query.
- <sensor type> is one of the following sensor types:

Sensor type	Description
current	Current sensor
voltage	Voltage sensor
activePower	Active power sensor
apparentPower	Apparent power sensor
powerFactor	Power factor sensor
activeEnergy	Active energy sensor

Sensor type	Description
lineFrequency	Line frequency sensor

Displayed information:

- Without the parameter "details," only the sensor reading, state, threshold, deassertion hysteresis and assertion timeout settings of the specified outlet sensor are displayed.
- With the parameter "details," more sensor information is displayed, including resolution and range.
- If the requested sensor type is not supported, the "Sensor is not available" message is displayed.

Inlet Sensor Threshold Information

This command syntax shows the specified inlet sensor's threshold-related information.

```
#       show sensor inlet <n> <sensor type>
```

To show detailed information, add the parameter "details" to the end of the command.

```
#       show sensor inlet <n> <sensor type> details
```

Variables:

- <n> is the number of the inlet whose sensors you want to query. For a single-inlet PDU, <n> is always the number 1.
- <sensor type> is one of the following sensor types:

Sensor type	Description
current	Current sensor
voltage	Voltage sensor
activePower	Active power sensor
apparentPower	Apparent power sensor
powerFactor	Power factor sensor
activeEnergy	Active energy sensor
unbalancedCurrent	Unbalanced load sensor
lineFrequency	Line frequency sensor

Displayed information:

- Without the parameter "details," only the reading, state, threshold, deassertion hysteresis and assertion timeout settings of the specified inlet sensor are displayed.
- With the parameter "details," more sensor information is displayed, including resolution and range.

- If the requested sensor type is not supported, the "Sensor is not available" message is displayed.

Environmental Sensor Threshold Information

This command syntax shows the specified environmental sensor's threshold-related information.

```
#       show sensor externalsensor <n>
```

To show detailed information, add the parameter "details" to the end of the command.

```
#       show sensor externalsensor <n> details


External sensor 3 (Temperature):
Reading: 31.8 deg C
State:   normal


Active Thresholds: Sensor specific thresholds


Default Thresholds for Temperature sensors:
Lower critical threshold: 10.0 deg C
Lower warning threshold:  15.0 deg C
Upper warning threshold:  30.0 deg C
Upper critical threshold: 35.0 deg C
Deassertion hysteresis:   1.0 deg C
Assertion timeout:       0 samples


Sensor Specific Thresholds:
Lower critical threshold: 8.0 deg C
Lower warning threshold:  13.0 deg C
Upper warning threshold:  28.0 deg C
Upper critical threshold: 33.0 deg C
Deassertion hysteresis:   1.0 deg C
Assertion timeout:       0 samples
```

Variables:

- <n> is the environmental sensor number. The environmental sensor number is the ID number assigned to the sensor, which can be found on the Peripherals page of the PDU web interface.

Displayed information:

- Without the parameter "details," only the reading, threshold, deassertion hysteresis and assertion timeout settings of the specified environmental sensor are displayed.
- With the parameter "details," more sensor information is displayed, including resolution and range.

Note: For a state sensor, the threshold-related and accuracy-related data is NOT available.

Environmental Sensor Default Thresholds

This command syntax shows a certain sensor type's default thresholds, which are the initial thresholds applying to the specified type of sensor.

```
#       show defaultThresholds <sensor type>
```

To show detailed information, add the parameter "details" to the end of the command.

```
#       show defaultThresholds <sensor type> details
```

Variables:

- <sensor type> is one of the following numeric sensor types:

Sensor types	Description
absoluteHumidity	Absolute humidity sensors
relativeHumidity	Relative humidity sensors
temperature	Temperature sensors
airPressure	Air pressure sensors
airFlow	Air flow sensors
vibration	Vibration sensors
all -- OR -- [blank]	All of the above numeric sensors

Displayed information:

- Without the parameter "details," only the default upper and lower thresholds, deassertion hysteresis and assertion timeout settings of the specified sensor type are displayed.
- With the parameter "details," the threshold range is displayed in addition to default thresholds settings.

Security Settings

This command shows the security settings of the PDU.

```
#       show security
```

To show detailed information, add the parameter "details" to the end of the command.

```
#       show security details
```

Displayed information:

- Without the parameter "details," the information including IP access control, role-based access control, password policy, and HTTPS encryption is displayed.
- With the parameter "details," more security information is displayed, such as user blocking time, user idle timeout and front panel permissions (if supported by your model).

Existing User Profiles

This command shows the data of one or all existing user profiles.

```
#       show user <user_name>
```

To show detailed information, add the parameter "details" to the end of the command.

```
#       show user <user_name> details
```

Variables:

- <user_name> is the name of the user whose profile you want to query. The variable can be one of the options: *all* or a user's name.

Option	Description
all -- OR -- [blank]	This option shows all existing user profiles.
a specific user's name	This option shows the profile of the specified user only.

Displayed information:

- Without the parameter "details," only four pieces of user information are displayed: user name, user "Enabled" status, SNMP v3 access privilege, and role(s).
- With the parameter "details," more user information is displayed, such as the telephone number, email address, preferred measurement units and so on.

Existing Roles

This command shows the data of one or all existing roles.

```
#       show roles <role_name>
```

Variables:

- <role_name> is the name of the role whose permissions you want to query. The variable can be one of the following options:

Option	Description
all -- OR -- [blank]	This option shows all existing roles.
a specific role's name	This option shows the data of the specified role only.

Displayed information:

- Role settings are displayed, including the role description and privileges.

Load Shedding Settings

This command shows the load shedding settings.

```
#       show loadshedding
```

Displayed information:

- The load shedding state is displayed along with non-critical outlets.

Note: The load shedding mode is associated with critical and non-critical outlets. To specify critical and non-critical outlets through CLI, see ***Specifying Non-Critical Outlets*** *(on page 45).*

Serial Port Settings

This command shows the baud rate setting of the serial port labeled CONSOLE on the PDU.

```
#       show serial
```

Event Log

The command used to show the event log begins with `show eventlog`. You can add either the *limit* or *class* parameters or both to show specific events.

- **Show the last 30 entries:**

```
#       show eventlog
```

- **Show a specific number of last entries in the event log:**

```
#       show eventlog limit <n>
```

- **Show a specific type of events only:**

```
#       show eventlog class <event_type>
```

- **Show a specific number of last entries associated with a specific type of events only:**

```
#       show eventlog limit <n> class <event_type>
```

Variables:

- <n> is one of the options: *all* or a number.

Option	Description
all	Displays all entries in the event log.
An integer number	Displays the specified number of last entries in the event log. The number ranges between 1 to 10,000.

- <event_type> is one of the following event types.

Event type	Description
all	All events.
device	Device-related events, such as system starting or firmware upgrade event.
userAdministration	User management events, such as a new user profile or a new role.
userActivity	User activities, such as login or logout.
pdu	Displays PDU-related events, such as entry or exit of the load shedding mode.
sensor	Internal or external sensor events, such as state changes of any sensors.
componentReachability	Component-monitoring records, such as a component being declared reachable or unreachable.
timerEvent	Scheduled action events.
surgeEvent	Surges detected by the device.

Wireless LAN Diagnostic Log

This command shows the diagnostic log for the wireless LAN connection.

```
#       show wlanlog
```

Component Reachability Information

This command shows all component reachability information with a list of monitored servers and status.

```
#       show componentReachability
```

Component Reachability Information for a Specific Component

To show the component reachability information for a certain device only, use the following command.

```
#       show componentReachability component <n>
```

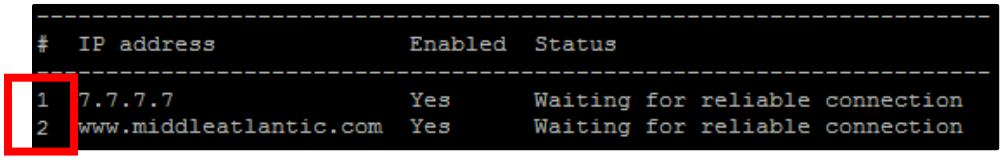
To show detailed information, add the parameter "details" to the end of the command.

```
#       show componentReachability component <n>
        details
```

Variables:

- <n> is a number representing the sequence of the device in the monitored component list.

  You can find each device's sequence number using the CLI command of `show componentReachability` as illustrated below.



Displayed information:

- Without the parameter "details," only the specified device's IP address, monitoring enabled/disabled state and current status are displayed.
- With the parameter "details," more settings for the specified device are displayed, such as number of pings and wait time prior to the next ping.

Command History

This command syntax shows the command history for current connection session.

```
#       show history
```

Displayed information:

- A list of commands that were previously entered in the current session is displayed.

History Buffer Length

This command syntax shows the length of the history buffer for storing history commands.

```
#       show history bufferlength
```

Displayed information:

- The current history buffer length is displayed.

Reliability Data

This command shows the reliability data.

```
#       show reliability data
```

Reliability Error Log

This command shows the reliability error log.

```
#       show reliability errorlog <n>
```

Variables:

- <n> is one of the options: *0* (zero) or any other integer number.

Option	Description
0 -- OR -- [blank]	Displays all entries in the reliability error log.
A specific integer number	Displays the specified number of last entries in the reliability error log.

Reliability Hardware Failures

This command shows a list of detected hardware failures.

```
#       show reliability hwfailures
```

For information, refer to ***Viewing Hardware Failures*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.

Examples

This section provides examples of the show command.

Example 1 - Basic Security Information

The diagram shows the output of the *show security* command.

```
# show security
IPv4 access control: Disabled

IPv6 access control: Disabled

Role based access control for IPv4: Disabled

Role based access control for IPv6: Disabled

Password aging: Disabled

Prevent concurrent user login:   No

Strong passwords: Disabled

Restricted Service Agreement: disabled
```

Example 2 - In-Depth Security Information

More information is displayed when typing the *show security details* command.

```
# show security details
IPv4 access control: Disabled

IPv6 access control: Disabled

Role based access control for IPv4: Disabled

Role based access control for IPv6: Disabled

Password aging: Disabled

Prevent concurrent user login:   No
Maximum number of failed logins: 3
User block time:                 10 minutes

User idle timeout: 10 minutes

Strong passwords: Disabled

Restricted Service Agreement: disabled
Restricted Service Agreement Banner Content:
Unauthorized access prohibited; all access and activities not explicitly authori
zed by management are unauthorized. All activities are monitored and logged. The
re is no privacy on this system. Unauthorized access and activities or any crimi
nal activity will be reported to appropriate authorities.

Front-Panel Permissions:
  Switch Outlet:             yes
  Switch Peripheral Actuator: no
```

Example 3 - Basic PDU Information

The diagram shows the output of the *show pdu* command.

```
# show pdu
PDU 'RackLink Premium'
Model:            RLNK-XXXXX
Firmware Version: 3.3.0.5-0
```

Example 4 - In-Depth PDU Information

More information is displayed when typing the *show pdu details* command. Displayed information varies depending on the model you purchased.

```
# show pdu details
PDU 'RackLink Premium'
Model:            RLNK-P920R
Firmware Version: 3.3.0.5-0
Serial Number:    855f53386dd85f82
Board Revision:   0x03

Relay behavior on power loss:    Non-latching
Default outlet state on startup: Last known state
Power cycle delay:               10 seconds

Outlet power sequence:       default
Outlet sequence delays:      1: 1.123 s
                             2-3: 0 s
                             4: 0.5 s
                             5: 1 s
                             6: 1.5 s
                             7: 2.5 s
                             8: 4 s
Inrush guard delay:          200 ms
Outlet initialization delay: 3 s

Voltage rating:   100-120V
Current rating:   16A
Frequency rating: 50/60Hz
Power rating:     1.6-1.9kVA

Internal beeper: Off

Sensor data retrieval:      Enabled
Measurements per log entry: 60

External sensor Z coordinate format: Rack units
Device altitude:                     0 ft
Peripheral Device Auto Management:   Enabled
```

Clearing Information

You can use the clear commands to remove unnecessary data from the PDU.

After typing a "clear" command, press Enter to execute it.

Note: Depending on your login name, the # prompt may be replaced by the > prompt. See ***Different CLI Modes and Prompts*** *(on page 20).*

Clearing Event Log

This command removes all data from the event log.

```
#       clear eventlog
```

-- OR --

```
#       clear eventlog /y
```

If you entered the command without "/y," a message appears, prompting you to confirm the operation. Type y to clear the event log or n to abort the operation.

If you type y, a message "Event log was cleared successfully" is displayed after all data in the event log is deleted.

Clearing WLAN Log

This command removes all data from the diagnostic log for the wireless LAN (WLAN) connection.

```
#       clear wlanlog
```

-- OR --

```
#       clear wlanlog /y
```

If you entered the command without "/y," a message appears, prompting you to confirm the operation. Type y to clear the WLAN log or n to abort the operation.

If you type y, a message "WLAN log was cleared successfully" is displayed to indicate all data in the WLAN log has been deleted.

Configuring the PDU and Network

To configure the PDU or network settings through the CLI, it is highly recommended to log in as the administrator so that you have full permissions.

To configure any settings, enter the configuration mode. Configuration commands are case sensitive so ensure you capitalize them correctly.

Entering Configuration Mode

Configuration commands function in configuration mode only.

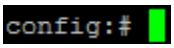
▶ **To enter configuration mode:**

1. Ensure you have entered administrator mode and the # prompt is displayed.

   Note: If you enter configuration mode from user mode, you may have limited permissions to make configuration changes. See ***Different CLI Modes and Prompts*** *(on page 20).*

2. Type `config` and press Enter.

   The config:# prompt appears, indicating that you have entered configuration mode.

3. Now you can type any configuration command and press Enter to change the settings.

Important: To apply new configuration settings, you must issue the "apply" command before closing the terminal emulation program. Closing the program does not save any configuration changes. See *Quitting Configuration Mode* (on page 42).

Quitting Configuration Mode

Both of "apply" and "cancel" commands let you quit the configuration mode. The difference is that "apply" saves all changes you made in the configuration mode while "cancel" aborts all changes.

▶ **To quit the configuration mode, use either command:**

```
config:#    apply
```

-- OR --

```
config:#    cancel
```

The # or > prompt appears after pressing Enter, indicating that you have entered the administrator or user mode. See ***Different CLI Modes and Prompts*** (on page 20).

PDU Configuration Commands

A PDU configuration command begins with *pdu*. You can use the PDU configuration commands to change the settings that apply to the whole PDU.

Changing the PDU Name

This command changes the PDU's name.

```
config:#   pdu name "<name>"
```

Variables:

- <name> is a string comprising up to 32 ASCII printable characters. The <name> variable must be enclosed in quotes when it contains spaces.

Setting the Outlet Relay Behavior

This command syntax determines the relay behavior of all outlets on your PDU.

```
config:#   pdu relayBehaviorOnPowerLoss <option>
```

Variables:

- <option> is one of the options: *latching* or *nonLatching*.

Note: For more information on the outlet relay behavior, refer to ***PDU Latching Relay Behavior*** *in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.*

Setting the Outlet Power-On Sequence

This command sets the outlet power-on sequence when the PDU powers up.

```
config:#   pdu outletSequence <option>
```

Variables:

- <option> is one of the options: *default*, or a comma-separated list of outlet numbers.

Option	Description
default	All outlets are switched ON in the ASCENDING order (from outlet 1 to the final outlet) when the PDU powers up.
A comma- separated list of outlet numbers	All outlets are switched ON in the order you specify using the comma-separated list. The list must include all outlets on the PDU.

*Note: Power-on sequencing is disabled in the latching mode. Refer to **PDU Latching Relay Behavior** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.*

Setting the Outlet Power-On Sequence Delay

This command sets the delays (in seconds) for outlets when turning on all outlets in sequence.

```
config:#   pdu outletSequenceDelay <outlet1>:<delay1>;<outlet2>:<delay2>;
           <outlet3>:<delay3>;...
```

Separate outlet numbers and their delay settings with a colon. Outlets followed by delays are separated with a semicolon.

Variables:

- <outlet1>, <outlet2>, <outlet3> and the like are individual outlet numbers or a range of outlets using a dash. For example, *3-8* represents outlets 3 to 8.
- <delay1>, <delay2>, <delay3> and the like are the delay time in seconds.

*Note: Power-on sequencing is disabled in the latching mode. Refer to **PDU Latching Relay Behavior** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.*

Setting the PDU-Defined Default Outlet State

This command determines the initial power condition of all outlets after powering up the PDU.

```
config:#   pdu outletStateOnDeviceStartup <option>
```

Variables:

- <option> is one of the options: *off*, *on* or *lastKnownState*.

Option	Description
off	Switches OFF all outlets when the PDU powers up.
on	Switches ON all outlets when the PDU powers up.
lastKnownState	Restores all outlets to the previous status before powering down the device when the PDU powers up again.

*Note: This feature does NOT take effect and cannot be configured on a PDU after the outlet relay is set to the "Latching" mode. Refer to **PDU Latching Relay Behavior** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.*

Setting the PDU-Defined Cycling Power-Off Period

This command sets the power-off period of the power cycling operation for all outlets.

```
config:#   pdu cyclingPowerOffPeriod <timing>
```

Variables:

- <timing> is the time of the cycling power-off period in seconds, which is an integer between 0 and 3600, or *pduDefined* for following the PDU-defined timing.

Setting the Inrush Guard Delay Time

This command sets the inrush guard delay.

```
config:#   pdu inrushGuardDelay <timing>
```

Variables:

- <timing> is a delay time between 100 and 100000 milliseconds.

Setting the Outlet Initialization Delay

This command determines the outlet initialization delay timing on device startup. For information on outlet initialization delay, refer to ***PDU*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.

```
config:#   pdu outletInitializationDelayOnDeviceStartup <timing>
```

Variables:

- <timing> is a delay time between 1 and 3600 seconds.

Note: This feature does NOT take effect and cannot be configured on a PDU after the outlet relay is set to the "Latching" mode. Refer to ***PDU Latching Relay Behavior*** *in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.*

Specifying Non-Critical Outlets

This command determines critical and non-critical outlets. It is associated with the load shedding mode. Refer to ***Load Shedding Mode*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.

```
config:#   pdu nonCriticalOutlets <outlets1>:false;<outlets2>:true
```

Separate outlet numbers and their settings with a colon. Separate each "false" and "true" setting with a semicolon.

Variables:

- <outlets1> is one or multiple outlet numbers to be set as critical outlets. Use commas to separate outlet numbers. Use a dash for a range of consecutive outlets. For example, *3-8* represents outlets 3 to 8.
- <outlets2> is one or multiple outlet numbers to be set as NON-critical outlets. Use commas to separate outlet numbers. Use a dash for a range of consecutive outlets. For example, *3-8* represents outlets 3 to 8.

Enabling or Disabling Data Logging

This command enables or disables the data logging feature.

```
config:#   pdu dataRetrieval <option>
```

Variables:

- <option> is one of the options: *enable* or *disable*.

Option	Description
enable	Enables the data logging feature.
disable	Disables the data logging feature.

For more information, refer to ***Setting Data Logging*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.

Setting Data Logging Measurements Per Entry

This command defines the number of measurements accumulated per log entry.

```
config:#   pdu measurementsPerLogEntry <number>
```

Variables:

- <number> is an integer between 1 and 600. The default is 60 samples per log entry.

For more information, refer to ***Setting Data Logging*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.

Setting the Z Coordinate Format for Environmental Sensors

This command enables or disables the use of rack units for specifying the height (Z coordinate) of environmental sensors.

```
config:#   pdu externalSensorsZCoordinateFormat <option>
```

Variables:

- <option> is one of the options: *rackUnits* or *freeForm*.

Option	Description
rackUnits	The height of the Z coordinate is measured in standard rack units. When this is selected, you can type a numeric value in the rack unit to describe the Z coordinate of any environmental sensors or dry contacts.
freeForm	Any alphanumeric string can be used for specifying the Z coordinate.

Note: After determining the format for the Z coordinate, you can set a value for it. See ***Setting the Z Coordinate*** *(on page 106).*

Enabling or Disabling Peripheral Device Auto Management

This command enables or disables the Peripheral Device Auto Management feature.

```
config:#   pdu peripheralDeviceAutoManagement <option>
```

Variables:

- <option> is one of the options: *enable* or *disable*.

Option	Description
enable	Enables the automatic management feature for environmental sensor packages.
disable	Disables the automatic management feature for environmental sensor packages.

For more information, refer to ***How the Automatic Management Function Works*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.

Examples

This section illustrates several PDU configuration examples.

Example 1 - PDU Naming

The following command assigns the name "my pdu" to the PDU.

```
config:#   pdu name "my pdu"
```

Example 2 - Outlet Sequence

The following command causes a 10-outlet PDU to first power on the 8th to 6th outlets and then the rest of outlets in the ascending order after the PDU powers up.

```
config:#   pdu outletSequence 8-6,1-5,9,10
```

Example 3 - Outlet Sequence Delay

The following command determines that the outlet 1's delay is 2.5 seconds, outlet 2's delay is 3 seconds, and the delay for outlets 3 through 5 is 10 seconds.

```
config:#   pdu outletSequenceDelay 1:2.5;2:3;3-5:10
```

Example 4 - Non-Critical Outlets

The following command sets outlets 1, 2, 3, 7, and 9 to be critical outlets, and 4, 5, 6, 8, 10, 11 and 12 to be non-critical outlets on a 12-outlet PDU.

```
config:#   pdu nonCriticalOutlets 1-3,7,9:false;4-6,8,10-12:true
```

Network Configuration Commands

A network configuration command begins with *network*. A number of network settings can be changed through the CLI, such as the IP address, transmission speed, duplex mode, and so on.

Configuring IPv4 Parameters

An IPv4 configuration command begins with *network ipv4*.

Setting the IPv4 Configuration Mode

This command determines the IP configuration mode.

```
config:#   network ipv4 interface <ETH> configMethod <mode>
```

Variables:

- <ETH> is one of the network interfaces: *ethernet (or ETH1/ETH2)*, *wireless* or *all*.

Interface	Description
eth1	Determine the IPv4 configuration mode of the ETH1 interface (wired networking).
eth2	Determine the IPv4 configuration mode of the ETH2 interface (wired networking).
wireless	Determine the IPv4 configuration mode of the WIRELESS interface (that is, wireless networking).

- <mode> is one of the modes: *dhcp* or *static*.

Mode	Description
dhcp	The IPv4 configuration mode is set to DHCP.
static	The IPv4 configuration mode is set to static IP address.

Setting the IPv4 Preferred Host Name

After selecting DHCP as the IPv4 configuration mode, you can specify the preferred host name, which is optional. The following is the command:

```
config:#   network ipv4 interface <ETH> preferredHostName <name>
```

Variables:

- <ETH> is one of the network interfaces: *ethernet (or ETH1/ETH2)*, *wireless* or *all*.

Interface	Description
eth1	Determine the IPv4 preferred host name of the ETH1 interface (that is, wired networking).
eth2	Determine the IPv4 preferred host name of the ETH2 interface (that is, wired networking).
wireless	Determine the IPv4 preferred host name of the WIRELESS interface (that is, wireless networking).

- <name> is a host name which:
  - Consists of alphanumeric characters and/or hyphens
  - Cannot begin or end with a hyphen
  - Cannot contain more than 63 characters
  - Cannot contain punctuation marks, spaces, and other symbols

Setting the IPv4 Address

After selecting the static IP configuration mode, you can use this command to assign a permanent IP address to the PDU.

```
config:#   network ipv4 interface <ETH> address <ip address>
```

Variables:

- <ETH> is one of the network interfaces: *ethernet (or ETH1/ETH2)*, *wireless* or *all*.

Interface	Description
eth1	Determine the IPv4 preferred host name of the ETH1 interface (that is, wired networking).

Interface	Description
eth2	Determine the IPv4 preferred host name of the ETH2 interface (that is, wired networking).
wireless	Determine the IPv4 address of the WIRELESS interface (that is, wireless networking).

- <ip address> is the IP address being assigned to your PDU. Its format is "IP address/prefix". For example, *192.168.84.99/32*.

Setting the IPv4 Gateway

After selecting the static IP configuration mode, you can use this command to specify the gateway.

```
config:#   network ipv4 gateway <ip address>
```

Variables:

- <ip address> is the IP address of the gateway. The value ranges from 0.0.0.0 to 255.255.255.255.

Setting IPv4 Static Routes

If the IPv4 network mode is set to static IP and your local network contains two subnets, you can configure static routes to enable or disable communications between the PDU and devices in the other subnet.

These commands are prefixed with *network ipv4 staticRoutes*.

Depending on whether the other network is directly reachable or not, there are two methods for adding a static route.
Method 1: add a static route when the other network is NOT directly reachable:

```
config:#   network ipv4 staticRoutes add <dest-1> <hop>
```

▶ **Method 2: add a static route when the other network is directly reachable:**

```
config:#   network ipv4 staticRoutes add <dest-1> interface <ETH>
```

▶ **Delete an existing static route:**

```
config:#   network ipv4 staticRoutes delete <route_ID>
```

▶ **Modify an existing static route:**

```
config:#   network ipv4 staticRoutes modify <route_ID> <dest-2> <hop>
```

-- OR --

```
config:#   network ipv4 staticRoutes modify <route_ID> <dest-2> interface <ETH>
```

Variables:

- <dest-1> is a combination of the IP address and subnet mask of the other subnet. The format is *IP address/subnet mask*.
- <hop> is the IP address of the next hop router.
- <ETH> is one of the interfaces: *ethernet (or ETH1/ETH2)*, and *wireless*.
- <route_ID> is the ID number of the route setting which you want to delete or modify.
- <dest-2> is a modified route setting that will replace the original route setting. Its format is *IP address/subnet mask*. You can modify either the IP address or the subnet mask or both.

Configuring IPv6 Parameters

An IPv6 configuration command begins with *network ipv6*.

Setting the IPv6 Configuration Mode

This command determines the IP configuration mode.

```
config:#   network ipv6 interface <ETH> configMethod <mode>
```

Variables:

- <ETH> is one of the network interfaces: *ethernet (or ETH1/ETH2)*, *wireless* or *all*.

Interface	Description
eth1	Determine the IPv6 configuration mode of the ETH1 interface (wired networking).
eth2	Determine the IPv6 configuration mode of the ETH2 interface (wired networking).
wireless	Determine the IPv6 configuration mode of the WIRELESS interface (that is, wireless networking).

- <mode> is one of the modes: *automatic* or *static*.

Mode	Description
automatic	The IPv6 configuration mode is set to automatic.
static	The IPv6 configuration mode is set to static IP address.

Setting the IPv6 Preferred Host Name

After selecting DHCP as the IPv6 configuration mode, you can specify the preferred host name, which is optional. The following is the command:

```
config:#   network ipv6 interface <ETH> preferredHostName <name>
```

Variables:

- <ETH> is one of the network interfaces: *ethernet (or ETH1/ETH2)*, *wireless* or *all*.

Interface	Description
eth1	Determine the IPv6 preferred host name of the ETH1 interface (wired networking).
eth2	Determine the IPv6 preferred host name of the ETH2 interface (wired networking).
wireless	Determine the IPv6 preferred host name of the WIRELESS interface (that is, wireless networking).

- <name> is a host name which:
  - Consists of alphanumeric characters and/or hyphens
  - Cannot begin or end with a hyphen
  - Cannot contain more than 63 characters
- Cannot contain punctuation marks, spaces, and other symbols

Setting the IPv6 Address

After selecting the static IP configuration mode, you can use this command to assign a permanent IP address to the PDU.

```
config:#          network ipv6 interface <ETH> address <ip address>
```

Variables:

- <ETH> is one of the network interfaces: *ethernet (or ETH1/ETH2)*, *wireless* or *all*.

Interface	Description
eth1	Determine the IPv6 address of the ETH1 interface (wired networking).
eth2	Determine the IPv6 address of the ETH2 interface (wired networking).
wireless	Determine the IPv6 address of the WIRELESS interface (that is, wireless networking).

- <ip address> is the IP address being assigned to your PDU. This value uses the IPv6 address format. Note that you must add */xx*, which indicates a prefix length of bits such as /64, to the end of this IPv6 address.

Setting the IPv6 Gateway

After selecting the static IP configuration mode, you can use this command to specify the gateway.

```
config:#   network ipv6 gateway <ip address>
```

Variables:

- <ip address> is the IP address of the gateway. This value uses the IPv6 address format.

Setting IPv6 Static Routes

If the IPv6 network mode is set to static IP and your local network contains two subnets, you can configure static routes to enable or disable communications between the PDU and devices in the other subnet.

These commands are prefixed with *network ipv6 staticRoutes*.

Depending on whether the other network is directly reachable or not, there are two methods for adding a static route.

▶ **Method 1: add a static route when the other network is NOT directly reachable:**

```
config:#   network ipv6 staticRoutes add <dest-1> <hop>
```

▶ **Method 2: add a static route when the other network is directly reachable:**

```
config:#   network ipv6 staticRoutes add <dest-1> interface <ETH>
```

▶ **Delete an existing static route:**

```
config:#   network ipv6 staticRoutes delete <route_ID>
```

▶ **Modify an existing static route:**

```
config:#   network ipv6 staticRoutes modify <route_ID> <dest-2> <hop>
```

-- OR --

```
config:#   network ipv6 staticRoutes modify <route_ID> <dest-2> interface <ETH>
```

Variables:

- <dest-1> is the IP address and prefix length of the subnet where the PDU belongs. The format is *IP address/prefix length*.
- <hop> is the IP address of the next hop router.
- <ETH> is one of the interfaces: *ethernet (or ETH1/ETH2)*, and *wireless*.
- <route_ID> is the ID number of the route setting which you want to delete or modify.
- <dest-2> is a modified route setting that will replace the original route setting. Its format is *IP address/prefix length*. You can modify either the IP address or the prefix length or both.

Configuring DNS Parameters

Use the following commands to configure DNS-related settings.

- **Specify the primary DNS server:**

```
config:#   network dns firstServer <ip address>
```

- **Specify the secondary DNS server:**

```
config:#   network dns secondServer <ip address>
```

- **Specify the third DNS server:**

```
config:#   network dns thirdServer <ip address>
```

- **Determine which IP address is used when the DNS server returns both IPv4 and IPv6 addresses:**

```
config:#   network dns resolverPreference <resolver>
```

Variables:

- <ip address> is the IP address of the DNS server.
- <resolver> is one of the options: *preferV4* or *preferV6.*

Option	Description
preferV4	Use the IPv4 addresses returned by the DNS server.
preferV6	Use the IPv6 addresses returned by the DNS server.

Setting LAN Interface Parameters

A LAN interface configuration command begins with *network ethernet*.

Enabling or Disabling the LAN Interface

This command enables or disables the LAN interface.

```
config:#   network ethernet <ETH> enabled <option>
```

Variables:

- <ETH> is *ethernet*.

Option	Description
eth1	ETH1 port.
eth2	ETH2 port.

- <option> is one of the options: *true or false*.

Option	Description
true	The specified network interface is enabled.
false	The specified network interface is disabled.

Changing the LAN Interface Speed

This command determines the LAN interface speed.

```
config:#   network ethernet <ETH> speed <option>
```

Variables:

- <ETH> is *ethernet*.

Option	Description
eth1	ETH1 port.
eth2	ETH2 port.

- <option> is one of the options: *auto*, *10Mbps*, *100Mbps or 1000Mbps*.

Option	Description
auto	System determines the optimum LAN speed through auto-negotiation.
10Mbps	The LAN speed is always 10 Mbps.
100Mbps	The LAN speed is always 100 Mbps.
1000Mbps	The LAN speed is always 1000 Mbps.

Changing the LAN Duplex Mode

This command determines the LAN interface duplex mode.

```
config:#   network ethernet <ETH> duplexMode <mode>
```

Variables:

- <ETH> is *ethernet*.

Option	Description
eth1	ETH1 port.
eth2	ETH2 port.

- <mode> is one of the modes: *auto*, *half* or *full*.

Option	Description
auto	The PDU selects the optimum transmission mode through auto-negotiation.
half	Half duplex: Data is transmitted in one direction (to or from the PDU) at a time.
full	Full duplex: Data is transmitted in both directions simultaneously.

Setting Wireless Parameters

You must configure wireless parameters, including Service Set Identifier (SSID), authentication method, Pre-Shared Key (PSK), and Basic Service Set Identifier (BSSID) after the wireless networking mode is enabled.

A wireless configuration command begins with *network wireless*.

Note: If current networking mode is not wireless, the SSID, PSK and BSSID values are not applied until the networking mode is changed to "wireless." In addition, a message appears, indicating that the active network interface is not wireless.

Setting the SSID

This command specifies the SSID string.

```
config:#   network wireless SSID <ssid>
```

Variables:

- <ssid> is the name of the wireless access point, which consists of:
  - Up to 32 ASCII characters
  - No spaces
  - ASCII codes 0x20 ~ 0x7E

Setting the Authentication Method

This command sets the wireless authentication method to either PSK or Extensible Authentication Protocol (EAP).

```
config:#   network wireless authMethod <method>
```

Variables:

- <method> is one of the authentication methods: *PSK* or *EAP*.

Method	Description
PSK	The wireless authentication method is set to PSK.
EAP	The wireless authentication method is set to EAP.

Setting the PSK

If the Pre-Shared Key (PSK) authentication method is selected, you must assign a PSK passphrase by using this command.

```
config:#   network wireless PSK <psk>
```

Variables:

- <psk> is a string or passphrase that consists of:

  - 8 to 63 characters

  - No spaces

  - ASCII codes 0x20 ~ 0x7E

Setting EAP Parameters

When the wireless authentication method is set to EAP, you must configure EAP authentication parameters, including outer authentication, inner authentication, EAP identity, password, and CA certificate.

- **Determine the outer authentication protocol:**

  ```
  config:#   network wireless eapOuterAuthentication <outer_auth>
  ```

- **Determine the inner authentication protocol:**

  ```
  config:#   network wireless eapInnerAuthentication <inner_auth>
  ```

- **Set the EAP identity:**

  ```
  config:#   network wireless eapIdentity <identity>
  ```

- **Set the EAP password:**

  ```
  config:#   network wireless eapPassword
  ```

After performing the above command, the PDU prompts you to enter the password. Then type the password and press Enter.

- **Provide a CA TLS certificate:**

```
config:#   network wireless eapCACertificate
```

After performing the above command, the system prompts you to enter the CA certificate's contents. For details, see ***EAP CA Certificate Example*** (on page 59).

- **Enable or disable verification of the TLS certificate chain:**

```
config:#   network wireless enableCertVerification <option1>
```

- **Allow expired and not yet valid TLS certificates:**

```
config:#   network wireless allowOffTimeRangeCerts <option2>
```

- **Allow wireless network connection with incorrect system time:**

```
config:#   network wireless allowConnectionWithIncorrectClock <option3>
```

Variables:

- The value of <outer_auth> is *PEAP* because the PDU only supports Protected Extensible Authentication Protocol (PEAP) as the outer authentication.
- The value of <inner_auth> is *MSCHAPv2* because the PDU only supports Microsoft's Challenge Authentication Protocol Version 2 (MSCHAPv2) as the inner authentication.
- <identity> is your user name for the EAP authentication.
- <option1> is one of the options: *true* or *false*.

Option	Description
true	Enables the verification of the TLS certificate chain.
false	Disables the verification of the TLS certificate chain.

- <option2> is one of the options: *true* or *false*.

Option	Description
true	Always make the wireless network connection successful even though the TLS certificate chain contains any certificate which is outdated or not valid yet.
false	The wireless network connection is NOT successfully established when the TLS certificate chain contains any certificate which is outdated or not valid yet.

- <option3> is one of the options: *true* or *false*.

Option	Description
true	Make the wireless network connection successful when the PDU system time is earlier than the firmware build before synchronizing with the NTP server, causing the TLS certificate to become invalid.
false	The wireless network connection is NOT successfully established when the PDU finds that the TLS certificate is not valid due to incorrect system time.

EAP CA Certificate Example

This section provides a CA certificate example only. Your CA certificate contents should be different from the contents displayed in this example.

To provide a CA certificate:

1. Make sure you have entered the configuration mode. See ***Entering Configuration Mode*** (on page 42).
2. Type the following command and press Enter.

   ```
   config:#   network wireless eapCACertificate
   ```

3. The system prompts you to enter the contents of the CA certificate.
4. Open a CA certificate using a text editor. You should see certificate contents similar to the following.

```
--- BEGIN CERTIFICATE ---
MIICjTCCAfigAwIBAgIEMaYgRzALBgkqhkiG9w0BAQQwRTELMAkGA1UEBhMCVVMx
NjA0BgNVBAoTLU5hdGlvbmFsIEFlcm9uYXV0aWNzIGFuZCBTcGFjZSBBZG1pbmlz
dHJhdGlvbjAmFxE5NjA1MjgxMzQ5MDUrMDgwMBcROTgwNTI4MTM0OTA1KzA4MDAw
ZzELMAkGA1UEBhMCVVMxNjA0BgNVBAoTLU5hdGlvbmFsIEFlcm9uYXV0aWNzIGFu
ZCBTcGFjZSBBZG1pbmlzdHJhdGlvbjEgMAkGA1UEBRMCMTYwEwYDVQQDEwxTdGV2
ZSBTY2hvY2gwWDALBgkqhkiG9w0BAQEDSQAwRgJBALrAwyYdgxmzNP/ts0Uyf6Bp
miJYktU/w4NG67ULaN4B5CnEz7k57s9o3YY3LecETgQ5iQHmkwlYDTL2fTgVfw0C
AQOjgaswgagwZAYDVR0ZAQH/BFowWDBWMFQxCzAJBgNVBAYTAlVTMTYwNAYDVQQK
Ey1OYXRpb25hbCBBZXJvbmF1dGljcyBhbmQgU3BhY2UgQWRtaW5pc3RyYXRpb24x
DTALBgNVBAMTBENSTDEwFwYDVR0BAQH/BA0wC4AJODMyOTcwODEwMBgGA1UdAgQR
MA8ECTgzMjk3MDgyM4ACBSAwDQYDVR0KBAYwBAMCBkAwCwYJKoZIhvcNAQEEA4GB
AH2y1VCEw/A4zaXzSYZJTTUi3uawbbFiS2yxHvgf28+8Js0OHXk1H1w2d6qOHH21
X82tZXd/0JtG0g1T9usFFBDvYK8O0ebgz/P5ELJnBL2+atObEuJy1ZZ0pBDWINR3
WkDNLCGiTkCKp0F5EWIrVDwh54NNevkCQRZita+z4IBO
--- END CERTIFICATE ---
```

5. Select and copy the contents as illustrated below, excluding the starting line containing "BEGIN CERTIFICATE" and the ending line containing "END CERTIFICATE."

```
MIICjTCCAfigAwIBAgIEMaYgRzALBgkqhkiG9w0BAQQwRTELMAkGA1UEBhMCVVMxNjA0BgNVBAoTLU5hdGlvbmF
sIEFlcm9uYXV0aWNzIGFuZCBTcGFjZSBBZG1pbmlzdHJhdGlvbjAmFxE5NjA1MjgxMzQ5MDUrMDgwMBcROTgwNT
I4MTM0OTA1KzA4MDAwZzELMAkGA1UEBhMCVVMxNjA0BgNVBAoTLU5hdGlvbmFsIEFlcm9uYXV0aWNzIGFuZCBTc
GFjZSBBZG1pbmlzdHJhdGlvbjEgMAkGA1UEBRMCMTYwEwYDVQQDEwxTdGV2ZSBTY2hvY2gwWDALBgkqhkiG9w0B
AQEDSQAwRgJBALrAwyYdgxmzNP/ts0Uyf6BpmiJYktU/w4NG67ULaN4B5CnEz7k57s9o3YY3LecETgQ5iQHmkwl
YDTL2fTgVfw0CAQOjgaswgagwZAYDVR0ZAQH/BFowWDBWMFQxCzAJBgNVBAYTAlVTMTYwNAYDVQQKEy1OYXRpb2
5hbCBBZXJvbmF1dGljcyBhbmQgU3BhY2UgQWRtaW5pc3RyYXRpb24xDTALBgNVBAMTBENSTDEwFwYDVR0BAQH/B
A0wC4AJODMyOTcwODEwMBgGA1UdAgQRMA8ECTgzMjk3MDgyM4ACBSAwDQYDVR0KBAYwBAMCBkAwCwYJKoZIhvcN
AQEEA4GBAH2y1VCEw/A4zaXzSYZJTTUi3uawbbFiS2yxHvgf28+8Js0OHXk1H1w2d6qOHH21X82tZXd/0JtG0g1
T9usFFBDvYK8O0ebgz/P5ELJnBL2+atObEuJy1ZZ0pBDWINR3WkDNLCGiTkCKp0F5EWIrVDwh54NNevkCQRZita
+z4IBO
```

6. Paste the contents in the terminal.
7. Press Enter.
8. Verify whether the system shows the following command prompt, indicating the provided CA certificate is valid.

```
config:#
```

Setting the BSSID

This command specifies the BSSID.

```
config:#   network wireless BSSID <bssid>
```

Variables:

- <bssid> is either the MAC address of the wireless access point or *none* for automatic selection.

Setting Network Service Parameters

A network service command begins with *network services*.

Setting the HTTP Port

The commands used to configure the HTTP port settings begin with *network services http*.

Change the HTTP port:

```
config:#   network services http port <n>
```

Enable or disable the HTTP port:

```
config:#   network services http enabled <option>
```

Variables:

- <n> is a TCP port number between 1 and 65535. The default HTTP port is 80.
- <option> is one of the options: *true* or *false*.

Option	Description
true	The HTTP port is enabled.
false	The HTTP port is disabled.

Setting the HTTPS Port

The commands used to configure the HTTPS port settings begin with *network services https*.

- **Change the HTTPS port:**

```
config:#   network services https port <n>
```

- **Enable or disable the HTTPS access:**

```
config:#   network services https enabled <option>
```

Variables:

- <n> is a TCP port number between 1 and 65535. The default HTTPS port is 443.
- <option> is one of the options: *true* or *false*.

Option	Description
true	Forces any access to the PDU via HTTP to be redirected to HTTPS.
false	No HTTP access is redirected to HTTPS.

Changing the Telnet Configuration

You can enable or disable the Telnet service, or change its TCP port using the CLI commands.

A Telnet command begins with *network services telnet*.

Enabling or Disabling Telnet

This command enables or disables the Telnet service.

```
config:#   network services telnet enabled <option>
```

Variables:

- <option> is one of the options: *true* or *false*.

Option	Description
true	The Telnet service is enabled.
false	The Telnet service is disabled.

Changing the Telnet Port

This command changes the Telnet port.

```
config:#   network services telnet port <n>
```

Variables:

- <n> is a TCP port number between 1 and 65535. The default Telnet port is 23.

Changing the SSH Configuration

You can enable or disable the SSH service, or change its TCP port using the CLI commands.

An SSH command begins with *network services ssh*.

Enabling or Disabling SSH

This command enables or disables the SSH service.

```
config:#   network services ssh enabled <option>
```

Variables:

- <option> is one of the options: *true* or *false*.

Option	Description
true	The SSH service is enabled.
false	The SSH service is disabled.

Changing the SSH Port

This command changes the SSH port.

```
config:#   network services ssh port <n>
```

Variables:

- <n> is a TCP port number between 1 and 65535. The default SSH port is 22.

Determining the SSH Authentication Method

This command syntax determines the SSH authentication method.

```
config:#   network services ssh authentication <auth_method>
```

Variables:

- <option> is one of the options: *passwordOnly*, *publicKeyOnly* or *passwordOrPublicKey*.

Option	Description
passwordOnly	Enables the password-based login only.
publicKeyOnly	Enables the public key-based login only.
passwordOrPublicKey	Enables both the password- and public key-based login. This is the default.

If the public key authentication is selected, you must enter a valid SSH public key for each user profile to log in over the SSH connection. See ***Specifying the SSH Public Key*** (on page 96).

Setting the SNMP Configuration

You can enable or disable the SNMP v1/v2c or v3 agent, configure the read and write community strings, or set the MIB-II parameters, such as sysContact, using the CLI commands.

An SNMP command begins with *network services snmp*.

Enabling or Disabling SNMP v1/v2c

This command enables or disables the SNMP v1/v2c protocol.

```
config:#   network services snmp v1/v2c <option>
```

Variables:

- <option> is one of the options: *enable* or *disable*.

Option	Description
enable	The SNMP v1/v2c protocol is enabled.
disable	The SNMP v1/v2c protocol is disabled.

Enabling or Disabling SNMP v3

This command enables or disables the SNMP v3 protocol.

```
config:#   network services snmp v3 <option>
```

Variables:

- <option> is one of the options: *enable* or *disable*.

Option	Description
enable	The SNMP v3 protocol is enabled.

Option	Description
disable	The SNMP v3 protocol is disabled.

Setting the SNMP Read Community

This command sets the SNMP read-only community string.

```
config:#   network services snmp readCommunity <string>
```

Variables:

- <string> is a string comprising 4 to 64 ASCII printable characters.
- The string CANNOT include spaces.

Setting the SNMP Write Community

This command sets the SNMP read/write community string.

```
config:#   network services snmp writeCommunity <string>
```

Variables:

- <string> is a string comprising 4 to 64 ASCII printable characters.
- The string CANNOT include spaces.

Setting the sysContact Value

This command sets the SNMP MIB-II sysContact value.

```
config:#   network services snmp sysContact <value>
```

Variables:

- <value> is a string comprising 0 to 255 alphanumeric characters.

Setting the sysName Value

This command sets the SNMP MIB-II sysName value.

```
config:#   network services snmp sysName <value>
```

Variables:

- <value> is a string comprising 0 to 255 alphanumeric characters.

Setting the sysLocation Value

This command sets the SNMP MIB-II sysLocation value.

```
config:#   network services snmp sysLocation <value>
```

Variables:

<value> is a string comprising 0 to 255 alphanumeric characters.

Changing the Modbus Configuration

You can enable or disable the Modbus agent, configure its read-only capability, or change its TCP port.

A Modbus command begins with *network services modbus*.

Enabling or Disabling Modbus

This command enables or disables the Modbus protocol.

```
config:#   network services modbus enabled <option>
```

Variables:

- <option> is one of the options: *true* or *false*.

Option	Description
true	The Modbus agent is enabled.
false	The Modbus agent is disabled.

Enabling or Disabling the Read-Only Mode

This command enables or disables the read-only mode for the Modbus agent.

```
config:#   network services modbus readonly <option>
```

Variables:

- <option> is one of the options: *true* or *false*.

Option	Description
true	The read-only mode is enabled.
false	The read-only mode is disabled.

Changing the Modbus Port

This command changes the Modbus port.

```
config:#   network services modbus port <n>
```

Variables:

- <n> is a TCP port number between 1 and 65535. The default Modbus port is 502.

Enabling or Disabling Service Advertising

This command enables or disables the zero configuration protocol, which enables advertising or auto discovery of network services. See *Chapter* 7: Enabling Service Advertising (on page 151) for details.

```
config:#   network services zeroconfig enabled <option>
```

Variables:

- <option> is one of the options: *true* or *false*.

Option	Description
true	The zero configuration protocol is enabled.
false	The zero configuration protocol is disabled.

Examples

This section illustrates several network configuration examples.

Example 1 - Networking Mode

The following command enables the wired networking mode.

```
config:#   network mode wired
```

Example 2 - Enabling Both IP Protocols

The following command determines that both IPv4 and IPv6 protocols are enabled.

```
config:#   network ip proto both
```

Example 3 - Wireless Authentication Method

The following command sets the wireless authentication method to PSK.

```
config:#   network wireless authMethod PSK
```

Example 4 - Static IPv4 Configuration

The following command enables the Static IP configuration mode.

```
config:#   network ipv4 ipConfigurationMode static
```

Time Configuration Commands

A time configuration command begins with *time*.

Determining the Time Setup Method

This command determines the method to configure the system date and time.

```
config:#   time method <method>
```

Variables:

- <method> is one of the time setup options: *manual* or *ntp*.

Mode	Description
manual	The date and time settings are customized.
ntp	The date and time settings synchronize with a specified NTP server.

Setting NTP Parameters

A time configuration command that is used to set the NTP parameters begins with *time ntp*.

Specifying the Primary NTP Server

This command specifies the primary time server if synchronization with the NTP server is enabled.

```
config:#   time ntp firstServer <first_server>
```

Variables:

- The <first_server> is the IP address or host name of the primary NTP server.

Specifying the Secondary NTP Server

This command specifies the primary time server if synchronization with the NTP server is enabled.

```
config:#   time ntp secondServer <second_server>
```

Variables:

- The <second_server> is the IP address or host name of the secondary NTP server.

Overriding DHCP-Assigned NTP Servers

This command determines whether the customized NTP server settings override the DHCP-specified NTP servers.

```
config:#   time ntp overrideDHCPProvidedServer <option>
```

Variables:

- <option> is one of these options: *true* or *false*.

Mode	Description
true	Customized NTP server settings override the DHCP-specified NTP servers.
false	Customized NTP server settings do NOT override the DHCP-specified NTP servers.

Deleting an NTP Server

The following commands delete the primary and/or secondary time server(s).

▶ **To delete the primary time server:**

```
config:#   time ntp firstServer ""
```

▶ **To delete the secondary time server:**

```
config:#   time ntp secondServer ""
```

Setting the Time Zone

The CLI has a list of time zones to configure the date and time for the PDU.

```
config:#   time zone
```

After a list of time zones is displayed, type the index number of the time zone or press Enter to cancel.

Example

▶ **To set the time zone:**

1. Type the time zone command as shown below and press Enter.
   ```
   config:#   time zone
   ```
2. The system shows a list of time zones. Type the index number of the desired time zone and press Enter.
3. Type `apply` for the selected time zone to take effect.

Customizing the Date and Time

If intending to manually configure the date and time, use the following CLI commands to specify them.

Note: You shall set the time configuration method to "manual" prior to customizing the date and time. See ***Determining the Time Setup Method*** *(on page 67).*

- **Assign the date:**

```
config:#   time set date <yyyy-mm-dd>
```

- **Assign the time:**

```
config:#   time set time <hh:mm:ss>
```

Variables:

Variable	Description
<yyyy-mm-dd>	Type the date in the format of yyyy-mm-dd. For example, type *2015-11-30* for November 30, 2015.
<hh:mm:ss>	Type the time in the format of hh:mm:ss in the 24-hour format. For example, type *13:50:20* for 1:50:20 pm.

Setting the Automatic Daylight Savings Time

This command determines whether the daylight savings time is applied to the time settings.

```
config:#   time autoDST <option>
```

Variables:

- <option> is one of the options: *enable* or *disable*.

Mode	Description
enable	Daylight savings time is enabled.
disable	Daylight savings time is disabled.

Examples

This section illustrates several time configuration examples.

Example 1 - Time Setup Method

The following command sets the date and time settings by using the NTP servers.

```
config:#   time method ntp
```

Example 2 - Primary NTP Server

The following command sets the primary time server to 192.168.80.66.

```
config:#   time ntp firstServer 192.168.80.66
```

Checking the Accessibility of NTP Servers

This command verifies the accessibility of NTP servers specified manually on your PDU and then shows the result. For instructions on specifying NTP servers via CLI, see ***Setting NTP Parameters*** (on page 67).

To perform this command successfully, you must:

- Own the "Change Date/Time Settings" permission.
- Customize NTP servers. See ***Setting NTP Parameters*** (on page 67).
- Make the customized NTP servers override the DHCP-assigned ones. See ***Overriding DHCP-Assigned NTP Servers*** (on page 67).

This command is available either in the administrator/user mode or in the configuration mode. See ***Different CLI Modes and Prompts*** (on page 20).

▶ **In the administrator/user mode:**

```
#          check ntp
```

▶ **In the configuration mode:**

```
config#    check ntp
```

Security Configuration Commands

A security configuration command begins with *security*.

Firewall Control

You can manage firewall control features through the CLI. The firewall control lets you set up rules that permit or disallow access to the PDU from a specific or a range of IP addresses.

- An IPv4 firewall configuration command begins with *security ipAccessControl ipv4*.
- An IPv6 firewall configuration command begins with *security ipAccessControl ipv6*.

Modifying Firewall Control Parameters

There are different commands for modifying firewall control parameters.

- *IPv4 commands*

Enable or disable the IPv4 firewall control feature:

```
config:#   security ipAccessControl ipv4 enabled <option>
```

Determine the default IPv4 firewall control policy for inbound traffic:

```
config:#   security ipAccessControl ipv4 defaultPolicyIn <policy>
```

Determine the default IPv4 firewall control policy for outbound traffic:

```
config:#   security ipAccessControl ipv4 defaultPolicyOut <policy>
```

- *IPv6 commands*

Enable or disable the IPv6 firewall control feature:

```
config:#   security ipAccessControl ipv6 enabled <option>
```

Determine the default IPv6 firewall control policy for inbound traffic:

```
config:#   security ipAccessControl ipv6 defaultPolicyIn <policy>
```

Determine the default IPv6 firewall control policy for outbound traffic:

```
config:#   security ipAccessControl ipv6 defaultPolicyOut <policy>
```

Variables:

- <option> is one of the options: *true* or *false*.

Option	Description
true	Enables the IP access control feature.
false	Disables the IP access control feature.

- <policy> is one of the options: *accept, drop* or *reject*.

Option	Description
accept	Accepts traffic from all IP addresses.
drop	Discards traffic from all IP addresses, without sending any failure notification to the source host.

Option	Description
reject	Discards traffic from all IP addresses, and an ICMP message is sent to the source host for failure notification.

Tip: You can combine both commands to modify all firewall control parameters at a time. See ***Multi-Command Syntax*** *(on page 122).*

Managing Firewall Rules

You can add, delete or modify firewall rules using the CLI commands.

- An IPv4 firewall control rule command begins with *security ipAccessControl ipv4 rule*.
- An IPv6 firewall control rule command begins with *security ipAccessControl ipv6 rule*.

Adding a Firewall Rule

Depending on where you want to add a new firewall rule in the list, the command for adding a rule varies.

- *IPv4 commands*

▶ **Add a new rule to the bottom of the IPv4 rules list:**

```
config:#   security ipAccessControl ipv4 rule add <direction> <ip_mask> <policy>
```

▶ **Add a new IPv4 rule by inserting it above or below a specific rule:**

```
config:#   security ipAccessControl ipv4 rule add <direction> <ip_mask> <policy>
           <insert> <rule_number>

           -- OR --

config:#   security ipAccessControl ipv4 rule add <direction> <insert> <rule_number>
           <ip_mask> <policy>
```

- *IPv6 commands*

▶ **Add a new rule to the bottom of the IPv6 rules list:**

```
config:#   security ipAccessControl ipv6 rule add <direction> <ip_mask> <policy>
```

▶ **Add a new IPv6 rule by inserting it above or below a specific rule:**

```
config:#   security ipAccessControl ipv6 rule add <direction> <ip_mask> <policy>
           <insert> <rule_number>

           -- OR --
```

```
config:#   security ipAccessControl ipv6 rule add <direction> <insert> <rule_number>
           <ip_mask> <policy>
```

Variables:

- <direction> is one of the options: *in* or *out*.

Direction	Description
in	Inbound traffic.
out	Outbound traffic.

- <ip_mask> is the combination of the IP address and subnet mask values (or prefix length), which are separated with a slash. For example, an IPv4 combination looks like this: *192.168.94.222/24*.
- <policy> is one of the options: *accept, drop* or *reject*.

Policy	Description
accept	Accepts traffic from/to the specified IP address(es).
drop	Discards traffic from/to the specified IP address(es), without sending any failure notification to the source or destination host.
reject	Discards traffic from/to the specified IP address(es), and an ICMP message is sent to the source or destination host for failure notification.

- <insert> is one of the options: *insertAbove* or *insertBelow*.

Option	Description
insertAbove	Inserts the new rule above the specified rule number. Then: *new rule's number = the specified rule number*
insertBelow	Inserts the new rule below the specified rule number. Then: *new rule's number = the specified rule number + 1*

- <rule_number> is the number of the existing rule which you want to insert the new rule above or below.

Modifying a Firewall Rule

Depending on what to modify in an existing rule, the command varies.

- *IPv4 commands*

Modify an IPv4 rule's IP address and/or subnet mask:

```
config:#   security ipAccessControl ipv4 rule modify <direction> <rule_number> ipMask
           <ip_mask>
```

Modify an IPv4 rule's policy:

```
config:#   security ipAccessControl ipv4 rule modify <direction> <rule_number> policy
           <policy>
```

Modify all contents of an existing IPv4 rule:

```
config:#   security ipAccessControl ipv4 rule modify <direction> <rule_number> ipMask
           <ip_mask> policy <policy>
```

- *IPv6 commands*

Modify an IPv6 rule's IP address and/or prefix length:

```
config:#   security ipAccessControl ipv6 rule modify <direction> <rule_number> ipMask
           <ip_mask>
```

Modify an IPv6 rule's policy:

```
config:#   security ipAccessControl ipv6 rule modify <direction> <rule_number> policy
           <policy>
```

Modify all contents of an IPv6 existing rule:

```
config:#   security ipAccessControl ipv6 rule modify <direction> <rule_number> ipMask
           <ip_mask> policy <policy>
```

Variables:

- <direction> is one of the options: *in* or *out*.

Direction	Description
in	Inbound traffic.
out	Outbound traffic.

- <rule_number> is the number of the existing rule that you want to modify.
- <ip_mask> is the combination of the IP address and subnet mask values (or prefix length), which are separated with a slash. For example, an IPv4 combination looks like this: *192.168.94.222/24*.
- <policy> is one of the options: *accept, drop* or *reject*.

Option	Description
accept	Accepts traffic from/to the specified IP address(es).
drop	Discards traffic from/to the specified IP address(es), without sending any failure notification to the source or destination host.
reject	Discards traffic from/to the specified IP address(es), and an ICMP message is sent to the source or destination host for failure notification.

Deleting a Firewall Rule

The following commands remove a specific IPv4 or IPv6 rule from the list.

- IPv4 commands

```
config:#   security ipAccessControl ipv4 rule delete <direction> <rule_number>
```

- IPv6 commands

```
config:#   security ipAccessControl ipv6 rule delete <direction> <rule_number>
```

Variables:

- <direction> is one of the options: *in* or *out*.

Direction	Description
in	Inbound traffic.
out	Outbound traffic.

- <rule_number> is the number of the existing rule that you want to remove.

Restricted Service Agreement

The CLI command used to set the Restricted Service Agreement feature begins with `security restrictedServiceAgreement`,

Enabling or Disabling the Restricted Service Agreement

This command activates or deactivates the Restricted Service Agreement.

```
config:#   security restrictedServiceAgreement enabled <option>
```

Variables:

- <option> is one of the options: *true* or *false*.

Option	Description
true	Enables the Restricted Service Agreement feature.
false	Disables the Restricted Service Agreement feature.

After the Restricted Service Agreement feature is enabled, the agreement's content is displayed in the login screen.

Do either of the following, or the login fails:

- In the web interface, select the checkbox labeled "I understand and accept the Restricted Service Agreement."

  Tip: To select the agreement checkbox using the keyboard, first press Tab to go to the checkbox and then Enter.

- In the CLI, type y when the confirmation message "I understand and accept the Restricted Service Agreement" is displayed.

Specifying the Agreement Contents

This command allows you to create or modify contents of the Restricted Service Agreement.

```
config:#   security restrictedServiceAgreement bannerContent
```

After performing the above command, do the following:

1. Type the text comprising up to 10,000 ASCII characters when the CLI prompts you to enter the content.
2. To end the content:
   a. Press Enter.
   b. Type `--END--` to indicate the end of the content.
   c. Press Enter again.

If the content is successfully entered, the CLI displays this message "Successfully entered Restricted Service Agreement" followed by the total number of entered characters in parentheses.

Note: The new content of Restricted Service Agreement is saved only after typing the `apply` command. See ***Quitting Configuration Mode*** *(on page 42).*

Example

The following example illustrates how to specify the content of the Restricted Service Agreement.

1. Type the following command and press Enter to start entering the content.

   ```
   config:#   security restrictedServiceAgreement bannerContent
   ```

2. Type the following content when the CLI prompts you to enter the content.

   ```
   IMPORTANT!! You are accessing a PDU. If you are not the system administrator, do NOT power
   off or power cycle any outlet without the permission of the system administrator.
   ```

3. Press Enter.
4. Type the following:
   `--END--`
5. Press Enter again.
6. Verify that the message "Successfully entered Restricted Service Agreement" is displayed, indicating that the content input is successful.

Login Limitation

The login limitation feature controls login-related limitations, such as password aging, simultaneous logins using the same user name, and the idle time permitted before forcing a user to log out.

A login limitation command begins with *security loginLimits*.

You can combine multiple commands to modify various login limitation parameters at a time. See ***Multi-Command Syntax*** (on page 122).

Single Login Limitation

This command enables or disables the single login feature, which controls whether multiple logins using the same login name simultaneously is permitted.

```
config:#   security loginLimits singleLogin <option>
```

Variables:

- <option> is one of the options: *enable* or *disable*.

Option	Description
enable	Enables the single login feature.
disable	Disables the single login feature.

Password Aging

This command enables or disables the password aging feature, which controls whether the password should be changed at a regular interval:

```
config:#   security loginLimits passwordAging <option>
```

Variables:

- <option> is one of the options: *enable* or *disable*.

Option	Description
enable	Enables the password aging feature.
disable	Disables the password aging feature.

Password Aging Interval

This command determines how often the password should be changed.

```
config:#   security loginLimits passwordAgingInterval <value>
```

Variables:

- <value> is a numeric value in days set for the password aging interval. The interval ranges from 7 to 365 days.

Idle Timeout

This command determines how long a user can remain idle before that user is forced to log out of the PDU web interface or CLI.

```
config:#   security loginLimits idleTimeout <value>
```

Variables:

- <value> is a numeric value in minutes set for the idle timeout. The timeout ranges from 1 to 1440 minutes (24 hours).

User Blocking

There are different commands for changing different user blocking parameters. These commands begin with `security userBlocking`.

You can combine multiple commands to modify the user blocking parameters at a time. See ***Multi-Command Syntax*** (on page 122).

- **Determine the maximum number of failed logins before blocking a user:**

  ```
  config:#   security userBlocking maximumNumberOfFailedLogins <value1>
  ```

- **Determine how long a user is blocked:**

  ```
  config:#   security userBlocking blockTime <value2>
  ```

Variables:

- <value1> is an integer between 3 and 10, or *unlimited*, which sets no limit on the maximum number of failed logins and thus disables the user blocking function.
- <value2> is a numeric value ranging from 1 to 1440 minutes (one day), or *infinite*, which blocks the user all the time until the user is unblocked manually.

Strong Passwords

The strong password commands determine whether a strong password is required for login, and what a strong password should contain at least.

A strong password command begins with `security strongPasswords`.

You can combine multiple strong password commands to modify different parameters at a time. See ***Multi-Command Syntax*** (on page 122).

Enabling or Disabling Strong Passwords

This command enables or disables the strong password feature.

```
config:#   security strongPasswords enabled <option>
```

Variables:

- <option> is one of the options: *true* or *false*.

Option	Description
true	Enables the strong password feature.
false	Disables the strong password feature.

Minimum Password Length

This command determines the minimum length of the password.

```
config:#   security strongPasswords minimumLength <value>
```

Variables:

- <value> is an integer between 8 and 32.

Maximum Password Length

This command determines the maximum length of the password.

```
config:#   security strongPasswords maximumLength <value>
```

Variables:

- <value> is an integer between 16 and 64.

Lowercase Character Requirement

This command determines whether a strong password includes at least a lowercase character.

```
config:#   security strongPasswords enforceAtLeastOneLowerCaseCharacter <option>
```

Variables:

- <option> is one of the options: *enable* or *disable*.

Option	Description
enable	At least one lowercase character is required.
disable	No lowercase character is required.

This command determines whether a strong password includes at least a uppercase character.

```
config:#   security strongPasswords enforceAtLeastOneUpperCaseCharacter <option>
```

Variables:

- <option> is one of the options: *enable* or *disable*.

Option	Description
enable	At least one uppercase character is required.
disable	No uppercase character is required.

Numeric Character Requirement

This command determines whether a strong password includes at least a numeric character.

```
config:#   security strongPasswords enforceAtLeastOneNumericCharacter <option>
```

Variables:

- <option> is one of the options: *enable* or *disable*.

Option	Description
enable	At least one numeric character is required.
disable	No numeric character is required.

Special Character Requirement

This command determines whether a strong password includes at least a special character.

```
config:#   security strongPasswords enforceAtLeastOneSpecialCharacter <option>
```

Variables:

- <option> is one of the options: *enable* or *disable*.

Option	Description
enable	At least one special character is required.
disable	No special character is required.

Maximum Password History

This command determines the number of previous passwords that CANNOT be repeated when changing the password.

```
config:#   security strongPasswords passwordHistoryDepth <value>
```

Variables:

- <value> is an integer between 1 and 12.

Role-Based Access Control

In addition to firewall access control based on IP addresses, you can configure other access control rules that are based on both IP addresses and users' roles.

- An IPv4 role-based access control command begins with *security roleBasedAccessControl ipv4*.
- An IPv6 role-based access control command begins with *security roleBasedAccessControl ipv6*.

Modifying Role-Based Access Control Parameters

There are different commands for modifying role-based access control parameters.

- *IPv4 commands*

▶ **Enable or disable the IPv4 role-based access control feature:**

```
config:#   security roleBasedAccessControl ipv4 enabled <option>
```

▶ **Determine the IPv4 role-based access control policy:**

```
config:#   security roleBasedAccessControl ipv4 defaultPolicy <policy>
```

- *IPv6 commands*

Enable or disable the IPv6 role-based access control feature:

```
config:#   security roleBasedAccessControl ipv6 enabled <option>
```

Determine the IPv6 role-based access control policy:

```
config:#   security roleBasedAccessControl ipv6 defaultPolicy <policy>
```

Variables:

- <option> is one of the options: *true* or *false*.

Option	Description
true	Enables the role-based access control feature.
false	Disables the role-based access control feature.

- <policy> is one of the options: *allow* or *deny*.

Policy	Description
allow	Accepts traffic from all IP addresses regardless of the user's role.
deny	Drops traffic from all IP addresses regardless of the user's role.

Tip: You can combine both commands to modify all role-based access control parameters at a time. See ***Multi-Command Syntax*** *(on page 122).*

Managing Role-Based Access Control Rules

You can add, delete or modify role-based access control rules.

- An IPv4 role-based access control command for managing rules begins with *security roleBasedAccessControl ipv4 rule*.
- An IPv6 role-based access control command for managing rules begins with *security roleBasedAccessControl ipv6 rule*.

Adding a Role-Based Access Control Rule

Depending on where you want to add a new rule in the list, the command syntax for adding a rule varies.

- *IPv4 commands*

Add a new rule to the bottom of the IPv4 rules list:

```
config:#  security roleBasedAccessControl ipv4 rule add <start_ip> <end_ip> <role>
          <policy>
```

Add a new IPv4 rule by inserting it above or below a specific rule:

```
config:#  security roleBasedAccessControl ipv4 rule add <start_ip> <end_ip> <role>
          <policy> <insert> <rule_number>
```

- *IPv6 commands*

Add a new rule to the bottom of the IPv6 rules list:

```
config:#  security roleBasedAccessControl ipv6 rule add <start_ip> <end_ip> <role>
          <policy>
```

Add a new IPv6 rule by inserting it above or below a specific rule:

```
config:#  security roleBasedAccessControl ipv6 rule add <start_ip> <end_ip> <role>
          <policy> <insert> <rule_number>
```

Variables:

- <start_ip> is the starting IP address.
- <end_ip> is the ending IP address.
- <role> is the role for which you want to create an access control rule.
- <policy> is one of the options: *allow* or *deny*.

Policy	Description
allow	Accepts traffic from the specified IP address range when the user is a member of the specified role
deny	Drops traffic from the specified IP address range when the user is a member of the specified role

- <insert> is one of the options: *insertAbove* or *insertBelow*.

Option	Description
insertAbove	Inserts the new rule above the specified rule number. Then: *new rule's number = the specified rule number*

Option	Description
insertBelow	Inserts the new rule below the specified rule number. Then: *new rule's number = the specified rule number + 1*

- <rule_number> is the number of the existing rule which you want to insert the new rule above or below.

Modifying a Role-Based Access Control Rule

Depending on what to modify in an existing rule, the command syntax varies.

- *IPv4 commands*

▶ **Modify a rule's IPv4 address range:**

```
config:#   security roleBasedAccessControl ipv4 rule modify <rule_number>
           startIpAddress <start_ip> endIpAddress <end_ip>
```

▶ **Modify an IPv4 rule's role:**

```
config:#   security roleBasedAccessControl ipv4 rule modify <rule_number> role
           <role>
```

▶ **Modify an IPv4 rule's policy:**

```
config:#   security roleBasedAccessControl ipv4 rule modify <rule_number> policy
           <policy>
```

▶ **Modify all contents of an existing IPv4 rule:**

```
config:#   security roleBasedAccessControl ipv4 rule modify <rule_number>
           startIpAddress <start_ip> endIpAddress <end_ip> role <role> policy
           <policy>
```

- *IPv6 commands*

▶ **Modify a rule's IPv6 address range:**

```
config:#   security roleBasedAccessControl ipv6 rule modify <rule_number>
           startIpAddress <start_ip> endIpAddress <end_ip>
```

▶ **Modify an IPv6 rule's role:**

```
config:#   security roleBasedAccessControl ipv6 rule modify <rule_number> role
           <role>
```

- **Modify an IPv6 rule's policy:**

```
config:#   security roleBasedAccessControl ipv6 rule modify <rule_number> policy
           <policy>
```

- **Modify all contents of an existing IPv6 rule:**

```
config:#   security roleBasedAccessControl ipv6 rule modify <rule_number>
           startIpAddress <start_ip> endIpAddress <end_ip> role <role> policy
           <policy>
```

Variables:

- <rule_number> is the number of the existing rule that you want to modify.
- <start_ip> is the starting IP address.
- <end_ip> is the ending IP address.
- <role> is one of the existing roles.
- <policy> is one of the options: *allow* or *deny*.

Policy	Description
allow	Accepts traffic from the specified IP address range when the user is a member of the specified role
deny	Drops traffic from the specified IP address range when the user is a member of the specified role

Deleting a Role-Based Access Control Rule

These commands remove a specific rule from the list.

- IPv4 commands

```
config:#   security roleBasedAccessControl ipv4 rule delete <rule_number>
```

- IPv6 commands

```
config:#   security roleBasedAccessControl ipv6 rule delete <rule_number>
```

Variables:

- <rule_number> is the number of the existing rule that you want to remove.

Enabling or Disabling Front Panel Outlet Switching

The following CLI commands control whether you can turn on or off an outlet by operating the front panel display.

- **To enable the front panel outlet control feature:**

```
config:#   security frontPanelPermissions add switchOutlet
```

- **To disable the front panel outlet control feature:**

```
config:#   security frontPanelPermissions remove switchOutlet
```

Enabling or Disabling Front Panel Actuator Control

The following CLI commands control whether you can turn on or off a connected actuator by operating the front panel display.

- **To enable the front panel actuator control feature:**

```
config:#   security frontPanelPermissions add switchActuator
```

- **To disable the front panel actuator control feature:**

```
config:#   security frontPanelPermissions remove switchActuator
```

Tip: If your PDU supports multiple front panel permissions, you can combine them into one command by adding a semicolon (;) between different permissions. For example, the following CLI command enables both front panel actuator control and outlet switching functions simultaneously.

```
security frontPanelPermissions add switchActuator;switchOutlet
```

Examples

This section illustrates several security configuration examples.

Example 1 - IPv4 Firewall Control Configuration

The following command sets up two parameters of the IPv4 access control feature.

```
config:#   security ipAccessControl ipv4 enabled true defaultPolicyIn accept
           defaultPolicyOut accept
```

Results:

- The IPv4 access control feature is enabled.
- The default policy for inbound traffic is set to "accept."
- The default policy for outbound traffic is set to "accept."

Example 2 - Adding an IPv4 Firewall Rule

The following command adds a new IPv4 access control rule and specifies its location in the list.

```
config:#   security ipAccessControl ipv4 rule add 192.168.84.123/24 accept
           insertAbove 5
```

Results:

- A new IPv4 firewall control rule is added to accept all packets sent from the IPv4 address 192.168.84.123.
- The newly added rule is inserted above the 5th rule. That is, the new rule becomes the 5th rule, and the original 5th rule becomes the 6th rule.

Example 3 - User Blocking

The following command sets up two user blocking parameters.

```
config:#   security userBlocking maximumNumberOfFailedLogins 5 blockTime 30
```

Results:

- The maximum number of failed logins is set to 5.
- The user blocking time is set to 30 minutes.

Example 4 - Adding an IPv4 Role-based Access Control Rule

The following command creates a newIPv4 role-based access control rule and specifies its location in the list.

```
config:#   security roleBasedAccessControl ipv4 rule add 192.168.78.50 192.168.90.100
           admin deny insertAbove 3
```

Results:

- A new IPv4 role-based access control rule is added, dropping all packets from any IPv4 address between 192.168.78.50 and 192.168.90.100 when the user is a member of the role "admin."
- The newly added IPv4 rule is inserted above the 3rd rule. That is, the new rule becomes the 3rd rule, and the original 3rd rule becomes the 4th rule.

Outlet Configuration Commands

An outlet configuration command begins with *outlet*. Such a command allows you to configure an individual outlet.

Changing the Outlet Name

This command names an outlet.

```
config:#   outlet <n> name "<name>"
```

Variables:

- <n> is the number of the outlet that you want to configure.
- <name> is a string comprising up to 32 ASCII printable characters. The <name> variable must be enclosed in quotes when it contains spaces.

Changing an Outlet's Default State

This command determines the initial power condition of an outlet after the PDU powers up.

```
config:#   outlet <n> stateOnDeviceStartup <option>
```

Variables:

- <n> is the number of the outlet that you want to configure.
- <option> is one of the options: *off*, *on, lastKnownState* and *pduDefined.*

Option	Description
off	Turn off the outlet.
on	Turn on the outlet.
lastKnownState	Restore the outlet to the state prior to last PDU power down.
pduDefined	PDU-defined setting.

Note: Setting the outlet's default state to an option other than pduDefined *overrides the PDU-defined default state on that outlet. See* ***Setting the PDU-Defined Default Outlet State*** *(on page 44).*

Setting an Outlet's Cycling Power-Off Period

This command determines the power-off period of the power cycling operation for a specific outlet.

```
config:#   outlet <n> cyclingPowerOffPeriod <timing>
```

Variables:

- <n> is the number of the outlet that you want to configure.
- <timing> is the time of the cycling power-off period in seconds, which is an integer between 0 and 3600, or *pduDefined* for following the PDU-defined timing.

Note: This setting overrides the PDU-defined cycling power-off period on a particular outlet. See ***Setting the PDU-Defined Cycling Power-Off Period*** *(on page 45).*

Example - Outlet Naming

The following command assigns the name "Win XP" to outlet 8.

```
config:#   outlet 8 name "Win XP"
```

Inlet Configuration Commands

An inlet configuration command begins with *inlet*. You can configure an inlet by using the inlet configuration command.

Changing the Inlet Name

This command syntax names an inlet.

```
config:#   inlet <n> name "<name>"
```

Variables:

- <n> is the number of the inlet that you want to configure. For a single-inlet PDU, <n> is always the number 1. The value is an integer between 1 and 50.
- <name> is a string comprising up to 32 ASCII printable characters. The <name> variable must be enclosed in quotes when it contains spaces.

Enabling or Disabling an Inlet (for Multi-Inlet PDUs)

Enabling or disabling an inlet takes effect on a multi-inlet PDU only.

This command enables or disables an inlet.

```
config:#   inlet <n> enabled <option>
```

Variables:

- <n> is the number of the inlet that you want to configure. For a single-inlet PDU, <n> is always the number 1. The value is an integer between 1 and 50.
- <option> is one of the options: *true* or *false*.

Option	Description
true	The specified inlet is enabled.
false	The specified inlet is disabled.

Note: If performing this command causes all inlets to be disabled, a warning message appears, prompting you to confirm. When this occurs, press y to confirm or n to cancel the operation.

Example - Inlet Naming

The following command assigns the name "AC source" to the inlet 1. If your PDU contains multiple inlets, this command names the 1st inlet.

```
config:#   inlet 1 name "AC source"
```

User Configuration Commands

Most user configuration commands begin with *user* except for the password change command.

Creating a User Profile

This command creates a new user profile.

```
config:#   user create <name> <option> <roles>
```

After performing the user creation command, the PDU prompts you to assign a password to the newly created user. Then:

1. Type the password and press Enter.
2. Re-type the same password for confirmation and press Enter.

Variables:

- <name> is a string comprising up to 32 ASCII printable characters. The <name> variable CANNOT contain spaces.
- <option> is one of the options: *enable* or *disable*.

Option	Description
enable	Enables the newly created user profile.
disable	Disables the newly created user profile.

- <roles> is a role or a list of comma-separated roles assigned to the specified user profile.

Modifying a User Profile

A user profile contains various parameters that you can modify.

Tip: You can combine all commands to modify the parameters of a specific user profile at a time. See ***Multi-Command Syntax*** *(on page 122).*

Changing a User's Password

This command allows you to change an existing user's password if you have the Administrator Privileges.

```
config:#   user modify <name> password
```

After performing the above command, PDU prompts you to enter a new password. Then:

1. Type a new password and press Enter.
2. Re-type the new password for confirmation and press Enter.

Variables:

- <name> is the name of the user whose settings you want to change.

Example

The following procedure illustrates how to change the password of the user "May."

1. Verify that you have entered the configuration mode. See ***Entering Configuration Mode*** (on page 42).
2. Type the following command to change the password for the user profile "May."

   ```
   config:#   user modify May password
   ```

3. Type a new password when prompted, and press Enter.
4. Type the same new password and press Enter.
5. If the password change is completed successfully, the config:# prompt appears.

Modifying a User's Personal Data

You can change a user's personal data, including the user's full name, telephone number, and email address.

Various commands can be combined to modify the parameters of a specific user profile at a time. See ***Multi-Command Syntax*** (on page 122).

- **Change a user's full name:**

  ```
  config:#   user modify <name> fullName "<full_name>"
  ```

- **Change a user's telephone number:**

  ```
  config:#   user modify <name> telephoneNumber "<phone_number>"
  ```

- **Change a user's email address:**

  ```
  config:#   user modify <name> eMailAddress <email_address>
  ```

Variables:

- <name> is the name of the user whose settings you want to change.
- <full_name> is a string comprising up to 32 ASCII printable characters. The <full_name> variable must be enclosed in quotes when it contains spaces.

- <phone_number> is the phone number that can reach the specified user. The <phone_number> variable must be enclosed in quotes when it contains spaces.
- <email_address> is the email address of the specified user.

Enabling or Disabling a User Profile

This command enables or disables a user profile. A user can log in to the PDU only after that user's profile is enabled.

```
config:#   user modify <name> enabled <option>
```

Variables:

- <name> is the name of the user whose settings you want to change.
- <option> is one of the options: *true* or *false*.

Option	Description
true	Enables the specified user profile.
false	Disables the specified user profile.

Forcing a Password Change

This command determines whether the password change is forced when a user logs in to the specified user profile next time.

```
config:#   user modify <name> forcePasswordChangeOnNextLogin <option>
```

Variables:

- <name> is the name of the user whose settings you want to change.
- <option> is one of the options: *true* or *false*.

Option	Description
true	A password change is forced on the user's next login.
false	No password change is forced on the user's next login.

Modifying SNMPv3 Settings

There are different commands to modify the SNMPv3 parameters of a specific user profile. You can combine all of the following commands to modify the SNMPv3 parameters at a time. See ***Multi-Command Syntax*** (on page 122).

▶ **Enable or disable the SNMP v3 access on the PDU for the specified user:**

```
config:#   user modify <name> snmpV3Access <option1>
```

Variables:

- <name> is the name of the user whose settings you want to change.
- <option1> is one of the options: *enable* or *disable*.

Option	Description
enable	Enables the SNMP v3 access permission for the specified user.
disable	Disables the SNMP v3 access permission for the specified user.

▶ **Determine the security level:**

```
config:#   user modify <name> securityLevel <option2>
```

Variables:

- <name> is the name of the user whose settings you want to change.
- <option2> is one of the options: *noAuthNoPriv*, *authNoPriv* or *authPriv*.

Option	Description
noAuthNoPriv	No authentication and no privacy.
authNoPriv	Authentication and no privacy.
authPriv	Authentication and privacy.

▶ **Determine whether the authentication passphrase is identical to the password:**

```
config:#   user modify <name> userPasswordAsAuthenticationPassphrase <option3>
```

Variables:

- <name> is the name of the user whose settings you want to change.
- <option3> is one of the options: *true* or *false*.

Option	Description
true	Authentication passphrase is identical to the password.
false	Authentication passphrase is different from the password.

▶ **Determine the authentication passphrase:**

```
config:#   user modify <name> authenticationPassPhrase <authentication_passphrase>
```

Variables:

- <name> is the name of the user whose settings you want to change.
- <authentication_passphrase> is a string used as an authentication passphrase, comprising 8 to 32 ASCII printable characters.

Determine whether the privacy passphrase is identical to the authentication passphrase:

```
config:#   user modify <name> useAuthenticationPassPhraseAsPrivacyPassPhrase
           <option4>
```

Variables:

- <name> is the name of the user whose settings you want to change.
- <option4> is one of the options: *true* or *false*.

Option	Description
true	Privacy passphrase is identical to the authentication passphrase.
false	Privacy passphrase is different from the authentication passphrase.

Determine the privacy passphrase:

```
config:#   user modify <name> privacyPassPhrase <privacy_passphrase>
```

Variables:

- <name> is the name of the user whose settings you want to change.
- <privacy_passphrase> is a string used as a privacy passphrase, comprising 8 to 32 ASCII printable characters.

Determine the authentication protocol:

```
config:#   user modify <name> authenticationProtocol <option5>
```

Variables:

- <name> is the name of the user whose settings you want to change.
- <option5> is one of the options: *MD5* or *SHA-1*.

Option	Description
MD5	MD5 authentication protocol is applied.
SHA-1	SHA-1 authentication protocol is applied.

Determine the privacy protocol:

```
config:#   user modify <name> privacyProtocol <option6>
```

Variables:

- <name> is the name of the user whose settings you want to change.
- <option6> is one of the options: *DES* or *AES-128*.

Option	Description
DES	DES privacy protocol is applied.
AES-128	AES-128 privacy protocol is applied.

Changing the Role(s)

This command changes the role(s) of a specific user.

```
config:#   user modify <name> roles <roles>
```

Variables:

- <name> is the name of the user whose settings you want to change.
- <roles> is a role or a list of comma-separated roles assigned to the specified user profile. See ***All Privileges*** (on page 101).

Changing Measurement Units

You can change the measurement units displayed for temperatures, length, and pressure for a specific user profile. Different measurement unit commands can be combined so that you can set all measurement units at a time. To combine all commands, see ***Multi-Command Syntax*** (on page 122).

Note: The measurement unit change only applies to the web interface and command line interface.

---

*Tip: To set the default measurement units applied to the PDU user interfaces for all users via CLI, see **Setting Default Measurement Units** (on page 98).*

Set the preferred temperature unit:

```
config:#   user modify <name> preferredTemperatureUnit <option1>
```

Variables:

- <name> is the name of the user whose settings you want to change.
- <option1> is one of the options: *C* or *F*.

Option	Description
C	This option displays the temperature in Celsius.
F	This option displays the temperature in Fahrenheit.

Set the preferred length unit:

```
config:#   user modify <name> preferredLengthUnit <option2>
```

Variables:

- <name> is the name of the user whose settings you want to change.
- <option2> is one of the options: *meter* or *feet*.

Option	Description
meter	This option displays the length or height in meters.
feet	This option displays the length or height in feet.

Set the preferred pressure unit:

```
config:#   user modify <name> preferredPressureUnit <option3>
```

Variables:

- <name> is the name of the user whose settings you want to change.
- <option3> is one of the options: *pascal* or *psi*.

Option	Description
pascal	This option displays the pressure value in Pascals (Pa).
psi	This option displays the pressure value in psi.

Specifying the SSH Public Key

If the SSH key-based authentication is enabled, specify the SSH public key for each user profile using the following procedure.

To specify or change the SSH public key for a specific user:

1. Type the SSH public key command as shown below and press Enter.

   ```
   config:#   user modify <name> sshPublicKey
   ```

2. The system prompts you to enter the contents of the SSH public key. Do the following to input the contents:
   a. Open your SSH public key with a text editor.
   b. Copy all contents in the text editor.
   c. Paste the contents into the terminal.
   d. Press Enter.

To remove an existing SSH public key:

1. Type the same command as shown above.
2. When the system prompts you to input the contents, press Enter without typing or pasting anything.

Example

The following procedure illustrates how to change the SSH public key for the user "assistant."

1. Verify that you have entered the configuration mode. See *Entering Configuration Mode* (on page 42).
2. Type the following command and press Enter.

   ```
   config:#   user modify assistant sshPublicKey
   ```

3. You are prompted to enter a new SSH public key.
4. Type the new key and press Enter.

Deleting a User Profile

This command deletes an existing user profile.

```
config:#   user delete <name>
```

Changing Your Own Password

Every user can change their own password via this command if they have the Change Own Password privilege. Note that this command does not begin with *user*.

```
config:#   password
```

After performing this command, the PDU prompts you to enter both current and new passwords respectively.

Important: After the password is changed successfully, the new password is effective immediately wether or not you type the command "apply" to save the changes.

Example

This procedure changes your own password:

1. Verify that you have entered the configuration mode. See ***Entering Configuration Mode*** (on page 42).
2. Type the following command and press Enter.
   ```
   config:#   password
   ```
3. Type the existing password and press Enter when the following prompt appears.
   ```
   Current password:
   ```
4. Type the new password and press Enter when the following prompt appears.
   ```
   Enter new password:
   ```
5. Re-type the new password for confirmation and press Enter when the following prompt appears.
   ```
   Re-type new password:
   ```

Setting Default Measurement Units

Default measurement units, including temperature, length, and pressure units, apply to the PDU user interfaces across all users except for those whose preferred measurement units are set differently by themselves or the administrator. Diverse measurement unit commands can be combined so that you can set all default measurement units at a time. To combine all commands, see ***Multi-Command Syntax*** (on page 122).

Note: The measurement unit change only applies to the web interface and command line interface.

Tip: To change the preferred measurement units displayed in the PDU user interfaces for a specific user via CLI, see ***Changing Measurement Units*** *(on page 95)*.

- **Set the default temperature unit:**

  ```
  config:#   user defaultpreferences preferredTemperatureUnit <option1>
  ```

Variables:

- <option1> is one of the options: *C* or *F*.

Option	Description
C	This option displays the temperature in Celsius.
F	This option displays the temperature in Fahrenheit.

Set the default length unit:

```
config:#   user defaultpreferences preferredLengthUnit <option2>
```

Variables:

- <option2> is one of the options: *meter* or *feet*.

Option	Description
meter	This option displays the length or height in meters.
feet	This option displays the length or height in feet.

Set the default pressure unit:

```
config:#   user defaultpreferences preferredPressureUnit <option3>
```

Variables:

- <option3> is one of the options: *pascal* or *psi*.

Option	Description
pascal	This option displays the pressure value in Pascals (Pa).
psi	This option displays the pressure value in psi.

Examples

This section illustrates several user configuration examples.

Example 1 - Creating a User Profile

The following command creates a new user profile and sets two parameters for the new user.

```
config:#   user create May enable admin
```

Results:

- A new user profile "May" is created.
- The new user profile is enabled.
- The **admin** role is assigned to the new user profile.

Example 2 - Modifying a User's Roles

The following command assigns two roles to the user "May."

```
config:#   user modify May roles admin,tester
```

Results:

- The user May has the union of all privileges of "admin" and "tester."

Example 3 - Default Measurement Units

The following command sets all default measurement units at a time.

```
config:#   user defaultpreferences preferredTemperatureUnit F preferredLengthUnit feet
           preferredPressureUnit psi
```

Results:

- The default temperature unit is set to Fahrenheit.
- The default length unit is set to feet.
- The default pressure unit is set to psi.

Role Configuration Commands

A role configuration command begins with *role*.

Creating a Role

This command creates a new role, with a list of semicolon-separated privileges assigned to the role.

```
config:#   role create <name> <privilege1>;<privilege2>;<privilege3>...
```

If a specific privilege contains any arguments, that privilege should be followed by a colon and the argument(s).

```
config:#   role create <name> <privilege1>:<argument1>,<argument2>...;
           <privilege2>:<argument1>,<argument2>...;
           <privilege3>:<argument1>,<argument2>...;
           ...
```

Variables:

- <name> is a string comprising up to 32 ASCII printable characters.
- <privilege1>, <privilege2>, <privilege3> and the like are names of the privileges assigned to the role. Separate each privilege with a semi-colon. See ***All Privileges*** (on page 101).
- <argument1>, <argument2> and the like are arguments set for a particular privilege. Separate a privilege and its argument(s) with a colon, and separate arguments with a comma if there are more than one argument for a privilege.

All Privileges

This table lists all privileges. Note that available privileges vary according to the model you purchased. For example, a PDU without the outlet switching function does not have the privilege "switchOutlet."

Privilege	Description
acknowledgeAlarms	Acknowledge Alarms
adminPrivilege	Administrator Privileges
changeAuthSettings	Change Authentication Settings
changeDataTimeSettings	Change Date/Time Settings
changeExternalSensorsConfiguration	Change Peripheral Device Configuration
changeModemConfiguration	Change Modem Configuration
changeNetworkSettings	Change Network Settings
changePassword	Change Own Password
changePduConfiguration	Change Pdu, Inlet, Outlet & Overcurrent Protector Configuration
changeSecuritySettings	Change Security Settings
changeSnmpSettings	Change SNMP Settings
changeUserSettings	Change Local User Management
clearLog	Clear Local Event Log
firmwareUpdate	Firmware Update
performReset	Reset (Warm Start)
switchOutlet*	Switch Outlet

Privilege	Description
switchActuator**	Switch Actuator
switchTransferSwitch	Switch Transfer Switch
viewEventSetup	View Event Settings
viewEverything	Unrestricted View Privileges
viewLog	View Local Event Log
viewSecuritySettings	View Security Settings
viewSnmpSettings	View SNMP Settings
viewUserSettings	View Local User Management

* The "switchOutlet" privilege requires an argument that is separated with a colon. The argument could be:

- All outlets, that is,
  `switchOutlet:all`
- An outlet number. For example:
  `switchOutlet:1`
  `switchOutlet:2`
  `switchOutlet:3`
- A list of comma-separated outlets. For example:
  `switchOutlet:1,3,5,7,8,9`

** The "switchActuator" privilege requires an argument that is separated with a colon. The argument could be:

- All actuators, that is,
  `switchActuator:all`
- An actuator's ID number. For example:
  `switchActuator:1`
  `switchActuator:2`
  `switchActuator:3`
- A list of comma-separated ID numbers of different actuators. For example:
  `switchActuator:1,3,6`

Note: The ID number of each actuator is shown in the PDU web interface. It is an integer between 1 and 32.

Modifying a Role

You can modify diverse parameters of an existing role, including its privileges.

Modify a role's description:

```
config:#   role modify <name> description "<description>"
```

Variables:

- <name> is a string comprising up to 32 ASCII printable characters.
- <description> is a description comprising alphanumeric characters. The <description> variable must be enclosed in quotes when it contains spaces.

Add more privileges to a specific role:

```
config:#   role modify <name> addPrivileges
           <privilege1>;<privilege2>;<privilege3>...
```

If a specific privilege contains any arguments, add a colon and the argument(s) after that privilege.

```
config:#   role modify <name> addPrivileges
           <privilege1>:<argument1>,<argument2>...;
           <privilege2>:<argument1>,<argument2>...;
           <privilege3>:<argument1>,<argument2>...;
           ...
```

Variables:

- <name> is a string comprising up to 32 ASCII printable characters.
- <privilege1>, <privilege2>, <privilege3> and the like are names of the privileges assigned to the role. Separate each privilege with a semi-colon. See ***All Privileges*** (on page 101).
- <argument1>, <argument2> and the like are arguments set for a particular privilege. Separate a privilege and its argument(s) with a colon, and separate arguments with a comma if there are more than one argument for a privilege.

Remove specific privileges from a role:

```
config:#   role modify <name> removePrivileges
           <privilege1>;<privilege2>;<privilege3>...
```

If a specific privilege contains any arguments, add a colon and the argument(s) after that privilege.

```
config:#   role modify <name> removePrivileges
           <privilege1>:<argument1>,<argument2>...;
           <privilege2>:<argument1>,<argument2>...;
           <privilege3>:<argument1>,<argument2>...;
           ...
```

Note: When removing privileges from a role, make sure the specified privileges and arguments (if any) exactly match those assigned to the role. Otherwise, the command fails to remove specified privileges that are not available.

Variables:

- <name> is a string comprising up to 32 ASCII printable characters.
- <privilege1>, <privilege2>, <privilege3> and the like are names of the privileges assigned to the role. Separate each privilege with a semi-colon. See ***All Privileges*** (on page 101).
- <argument1>, <argument2> and the like are arguments set for a particular privilege. Separate a privilege and its argument(s) with a colon, and separate arguments with a comma if there are more than one argument for a privilege.

Deleting a Role

This command deletes an existing role.

```
config:#   role delete <name>
```

Example - Creating a Role

The following command creates a new role and assigns privileges to the role.

```
config:#   role create tester firmwareUpdate;viewEventSetup
```

Results:

- A new role "tester" is created.
- Two privileges are assigned to the role: firmwareUpdate (Firmware Update) and viewEventSetup (View Event Settings).

Environmental Sensor Configuration Commands

An environmental sensor configuration command begins with *externalsensor*. You can configure the name and location parameters of an individual environmental sensor.

Note: To configure an actuator, see ***Actuator Configuration Commands*** *(on page 116).*

Changing the Sensor Name

This command names an environmental sensor.

```
config:#   externalsensor <n> name "<name>"
```

Variables:

- <n> is the ID number of the environmental sensor that you want to configure. The ID number is available in the PDU web interface or using the command "`show externalsensors <n>`" in the CLI. It is an integer between 1 and 32.
- <name> is a string comprising up to 32 ASCII printable characters. The <name> variable must be enclosed in quotes when it contains spaces.

Note: To name an actuator, see ***Actuator Configuration Commands*** *(on page 116).*

Specifying the CC Sensor Type

The contact closure sensor (RLNK-CONT) supports the connection of diverse third-party detectors/switches. You must specify the type of connected detector/switch for proper operation. Use this command when you need to specify the sensor type.

```
config:#   externalsensor <n> sensorSubType <sensor_type>
```

Variables:

- <n> is the ID number of the environmental sensor that you want to configure. The ID number is available in the PDU web interface or using the command "`show externalsensors <n>`" in the CLI. It is an integer between 1 and 32.
- <sensor_type> is one of these types: *contact*, *smokeDetection*, *waterDetection* or *vibration*.

Type	Description
contact	The connected detector/switch is for detection of door lock or door closed/open status.
smokeDetection	The connected detector/switch is for detection of the smoke presence.
waterDetection	The connected detector/switch is for detection of the water presence.
vibration	The connected detector/switch is for detection of the vibration.

Setting the X Coordinate

This command specifies the X coordinate of an environmental sensor.

```
config:#   externalsensor <n> xlabel "<coordinate>"
```

Variables:

- <n> is the ID number of the environmental sensor that you want to configure. The ID number is available in the PDU web interface or using the command "`show externalsensors <n>`" in the CLI. It is an integer between 1 and 32.
- <coordinate> is a string comprising up to 24 ASCII printable characters, and it must be enclosed in quotes.

Setting the Y Coordinate

This command specifies the Y coordinate of an environmental sensor.

```
config:#   externalsensor <n> ylabel "<coordinate>"
```

Variables:

- <n> is the ID number of the environmental sensor that you want to configure. The ID number is available in the PDU web interface or using the command "`show externalsensors <n>`" in the CLI. It is an integer between 1 and 32.
- <coordinate> is a string comprising up to 24 ASCII printable characters, and it must be enclosed in quotes.

Setting the Z Coordinate

This command specifies the Z coordinate of an environmental sensor.

```
config:#   externalsensor <n> zlabel "<coordinate>"
```

Variables:

- <n> is the ID number of the environmental sensor that you want to configure. The ID number is available in the PDU web interface or using the command "`show externalsensors <n>`" in the CLI. It is an integer between 1 and 32.
- Depending on the Z coordinate format you set, there are two types of values for the <coordinate> variable:

Type	Description
Free form	<coordinate> is a string comprising up to 24 ASCII printable characters, and it must be enclosed in quotes.
Rack units	<coordinate> is an integer number in rack units.

Note: To specify the Z coordinate using the rack units, see ***Setting the Z Coordinate Format for Environmental Sensors*** *(on page 46).*

Changing the Sensor Description

This command provides a description for a specific environmental sensor.

```
config:#   externalsensor <n> description "<description>"
```

Variables:

- <n> is the ID number of the environmental sensor that you want to configure. The ID number is available in the PDU web interface or using the command "`show externalsensors <n>`" in the CLI. It is an integer between 1 and 32.
- <description> is a string comprising up to 64 ASCII printable characters, and it must be enclosed in quotes.

Using Default Thresholds

This command determines whether default thresholds, including the deassertion hysteresis and assertion timeout, are applied to a specific environmental sensor.

```
config:#   externalsensor <n> useDefaultThresholds <option>
```

Variables:

- <n> is the ID number of the environmental sensor that you want to configure. The ID number is available in the PDU web interface or using the command "`show externalsensors <n>`" in the CLI. It is an integer between 1 and 32.
- <option> is one of the options: *true* or *false*.

Option	Description
true	Default thresholds are selected as the threshold option for the specified sensor.
false	Sensor-specific thresholds are selected as the threshold option for the specified sensor.

Setting the Alarmed to Normal Delay for DX-PIR

This command determines the value of the Alarmed to Normal Delay setting for a DX-PIR presence detector.

```
config:#   externalsensor <n> alarmedToNormalDelay <time>
```

Variables:

- <n> is the ID number of the environmental sensor that you want to configure. The ID number is available in the PDU web interface or using the command "`show externalsensors <n>`" in the CLI. It is an integer between 1 and 32.
- <time> is an integer number in seconds, ranging between 0 and 300.

Examples

This section illustrates several environmental sensor configuration examples.

Example 1 - Environmental Sensor Naming

The following command assigns the name "Cabinet humidity" to the environmental sensor with the ID number 4.

```
config:#   externalsensor 4 name "Cabinet humidity"
```

Example 2 - Sensor Threshold Selection

The following command sets the environmental sensor #1 to use the default thresholds, including the deassertion hysteresis and assertion timeout, as its threshold settings.

```
config:#   externalsensor 1 useDefaultThresholds true
```

Configuring Environmental Sensors' Default Thresholds

You can set the default values of upper and lower thresholds, deassertion hysteresis and assertion timeout on a sensor type basis, including temperature, humidity, air pressure and air flow sensors. The default thresholds automatically apply to all environmental sensors that are newly detected or added.

A default threshold configuration command begins with *defaultThresholds*.

You can configure various default threshold settings for the same sensor type at a time by combining multiple commands. See ***Multi-Command Syntax*** (on page 122).

- **Set the Default Upper Critical Threshold for a specific sensor type:**

```
config:#   defaultThresholds <sensor type> upperCritical <value>
```

- **Set the Default Upper Warning Threshold for a specific sensor type:**

```
config:#   defaultThresholds <sensor type> upperWarning <value>
```

- **Set the Default Lower Critical Threshold for a specific sensor type:**

```
config:#   defaultThresholds <sensor type> lowerCritical <value>
```

- **Set the Default Lower Warning Threshold for a specific sensor type:**

```
config:#   defaultThresholds <sensor type> lowerWarning <value>
```

- **Set the Default Deassertion Hysteresis for a specific sensor type:**

```
config:#   defaultThresholds <sensor type> hysteresis <hy_value>
```

- **Set the Default Assertion Timeout for a specific sensor type:**

```
config:#   defaultThresholds <sensor type> assertionTimeout <as_value>
```

Variables:

- <sensor type> is one of the following numeric sensor types:

Sensor types	Description
absoluteHumidity	Absolute humidity sensors
relativeHumidity	Relative humidity sensors
temperature	Temperature sensors

Sensor types	Description
airPressure	Air pressure sensors
airFlow	Air flow sensors
vibration	Vibration sensors

- <value> is the value for the specified threshold of the specified sensor type. Note that diverse sensor types use different measurement units.

Sensor types	Measurement units
absoluteHumidity	g/m^3 (that is, g/m³)
relativeHumidity	%
temperature	Degrees Celsius (°C) or Fahrenheit (°F), depending on your measurement unit settings.
airPressure	Pascal (Pa) or psi, depending on your measurement unit settings.
airFlow	m/s
vibration	g

- <hy_value> is the deassertion hysteresis value applied to the specified sensor type.
- <as_value> is the assertion timeout value applied to the specified sensor type. It ranges from 0 to 100 (samples).

Example - Default Upper Thresholds for Temperature

It is assumed that your preferred measurement unit for temperature is set to degrees Celsius. Then the following command sets the default Upper Warning threshold to 20°C and Upper Critical threshold to 24°C for all temperature sensors.

```
config:#       defaultThresholds temperature upperWarning 20 upperCritical 24
```

Sensor Threshold Configuration Commands

A sensor configuration command begins with *sensor*. You can use the commands to configure the threshold, hysteresis and assertion timeout values for any sensor associated with the following items:

- Outlets
- Inlets
- Overcurrent protectors

- Environmental sensors

It is permitted to assign a new value to the threshold at any time regardless of whether the threshold has been enabled.

Commands for Outlet Sensors

A sensor configuration command for outlets begins with *sensor outlet*.

You can configure various outlet sensor threshold settings at a time by combining multiple commands. See *Multi-Command Syntax* (on page 122).

▶ **Set the Upper Critical threshold for an outlet sensor:**

```
config:#   sensor outlet <n> <sensor type> upperCritical <option>
```

▶ **Set the Upper Warning threshold for an outlet sensor:**

```
config:#   sensor outlet <n> <sensor type> upperWarning <option>
```

▶ **Set the Lower Critical threshold for an outlet sensor:**

```
config:#   sensor outlet <n> <sensor type> lowerCritical <option>
```

▶ **Set the Lower Warning threshold for an outlet sensor:**

```
config:#   sensor outlet <n> <sensor type> lowerWarning <option>
```

▶ **Set the deassertion hysteresis for an outlet sensor:**

```
config:#   sensor outlet <n> <sensor type> hysteresis <hy_value>
```

▶ **Set the assertion timeout for an outlet sensor:**

```
config:#   sensor outlet <n> <sensor type> assertionTimeout <as_value>
```

Variables:

- <n> is the number of the outlet that you want to configure.
- <sensor type> is one of the following sensor types:

Sensor type	Description
current	Current sensor
voltage	Voltage sensor
activePower	Active power sensor

Sensor type	Description
apparentPower	Apparent power sensor
powerFactor	Power factor sensor
activeEnergy	Active energy sensor
lineFrequency	Line frequency sensor

Note: If the requested sensor type is not supported, the "Sensor is not available" message is displayed.

- <option> is one of the options: *enable*, *disable* or a numeric value.

Option	Description
enable	Enables the specified threshold for a specific outlet sensor.
disable	Disables the specified threshold for a specific outlet sensor.
A numeric value	Sets a value for the specified threshold of a specific outlet sensor and enables this threshold at the same time.

- <hy_value> is a numeric value that is assigned to the hysteresis for the specified outlet sensor. Refer to ***"To De-assert" and Deassertion Hysteresis*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.
- <as_value> is a number in samples that is assigned to the assertion timeout for the specified outlet sensor. Refer to ***Configuring An Assertion Timeout*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.

Commands for Inlet Sensors

A sensor configuration command for inlets begins with *sensor inlet*.

You can configure various inlet sensor threshold settings at a time by combining multiple commands. See ***Multi-Command Syntax*** (on page 122).

▶ **Set the Upper Critical threshold for an inlet sensor:**

```
config:#   sensor inlet <n> <sensor type> upperCritical <option>
```

▶ **Set the Upper Warning threshold for an inlet sensor:**

```
config:#   sensor inlet <n> <sensor type> upperWarning <option>
```

Set the Lower Critical threshold for an inlet sensor:

```
config:#   sensor inlet <n> <sensor type> lowerCritical <option>
```

Set the Lower Warning threshold for an inlet sensor:

```
config:#   sensor inlet <n> <sensor type> lowerWarning <option>
```

Set the deassertion hysteresis for an inlet sensor:

```
config:#   sensor inlet <n> <sensor type> hysteresis <hy_value>
```

Set the assertion timeout for an inlet sensor:

```
config:#   sensor inlet <n> <sensor type> assertionTimeout <as_value>
```

Variables:

- <n> is the number of the inlet that you want to configure. For a single-inlet PDU, <n> is always the number 1.
- <sensor type> is one of the following sensor types:

Sensor type	Description
current	Current sensor
peakCurrent	Peak current sensor
voltage	Voltage sensor
activePower	Active power sensor
apparentPower	Apparent power sensor
powerFactor	Power factor sensor
activeEnergy	Active energy sensor
unbalancedCurrent	Unbalanced load sensor
lineFrequency	Line frequency sensor
residualCurrent	Residual current sensor
phaseAngle	Inlet phase angle sensor

Note: If the requested sensor type is not supported, the "Sensor is not available" message is displayed.

- <option> is one of the options: *enable*, *disable* or a numeric value.

Option	Description
enable	Enables the specified threshold for a specific inlet sensor.
disable	Disables the specified threshold for a specific inlet sensor.
A numeric value	Sets a value for the specified threshold of a specific inlet sensor and enables this threshold at the same time.

- <hy_value> is a numeric value that is assigned to the hysteresis for the specified inlet sensor. Refer to ***"To De-assert" and Deassertion Hysteresis*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.
- <as_value> is a numeric value that is assigned to the assertion timeout for the specified inlet sensor. Refer to ***Configuring An Assertion Timeout*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.

Commands for Overcurrent Protector Sensors

A sensor configuration command for overcurrent protectors begins with *sensor ocp*.

You can configure various overcurrent protector threshold settings at a time by combining multiple commands. See ***Multi-Command Syntax*** (on page 122).

- **Set the Upper Critical threshold for an overcurrent protector:**

```
config:#   sensor ocp <n> <sensor type> upperCritical <option>
```

- **Set the Upper Warning threshold for an overcurrent protector:**

```
config:#   sensor ocp <n> <sensor type> upperWarning <option>
```

- **Set the Lower Critical threshold for an overcurrent protector:**

```
config:#   sensor ocp <n> <sensor type> lowerCritical <option>
```

- **Set the Lower Warning threshold for an overcurrent protector:**

```
config:#   sensor ocp <n> <sensor type> lowerWarning <option>
```

- **Set the deassertion hysteresis for an overcurrent protector:**

```
config:#   sensor ocp <n> <sensor type> hysteresis <hy_value>
```

Set the assertion timeout for an overcurrent protector:

```
config:#   sensor ocp <n> <sensor type> assertionTimeout <as_value>
```

Variables:

- <n> is the number of the overcurrent protector that you want to configure.
- <sensor type> is one of the following sensor types:

Sensor type	Description
current	Current sensor

Note: If the requested sensor type is not supported, the "Sensor is not available" message is displayed.

- <option> is one of the options: *enable*, *disable* or a numeric value.

Option	Description
enable	Enables the specified threshold for the overcurrent protector sensor.
disable	Disables the specified threshold for the overcurrent protector sensor.
A numeric value	Sets a value for the specified threshold of the overcurrent protector sensor and enables this threshold at the same time.

- <hy_value> is a numeric value that is assigned to the hysteresis for the specified overcurrent protector sensor. Refer to ***"To De-assert" and Deassertion Hysteresis*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.
- <as_value> is a number in samples that is assigned to the assertion timeout for the specified overcurrent protector sensor. Refer to ***Configuring An Assertion Timeout*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.

Commands for Environmental Sensors

A sensor threshold configuration command for environmental sensors begins with *sensor externalsensor*.

You can configure various environmental sensor threshold settings at a time by combining multiple commands. See ***Multi-Command Syntax*** (on page 122).

Set the Upper Critical threshold for an environmental sensor:

```
config:#   sensor externalsensor <n> <sensor type> upperCritical <option>
```

- **Set the Upper Warning threshold for an environmental sensor:**

```
config:#   sensor externalsensor <n> <sensor type> upperWarning <option>
```

- **Set the Lower Critical threshold for an environmental sensor:**

```
config:#   sensor externalsensor <n> <sensor type> lowerCritical <option>
```

- **Set the Lower Warning threshold for an environmental sensor:**

```
config:#   sensor externalsensor <n> <sensor type> lowerWarning <option>
```

- **Set the deassertion hysteresis for an environmental sensor:**

```
config:#   sensor externalsensor <n> <sensor type> hysteresis <hy_value>
```

- **Set the assertion timeout for an environmental sensor:**

```
config:#   sensor externalsensor <n> <sensor type> assertionTimeout <as_value>
```

Variables:

- <n> is the ID number of the environmental sensor that you want to configure. The ID number is available in the PDU web interface or using the command "`show externalsensors <n>`" in the CLI. It is an integer between 1 and 32.
- <sensor type> is one of these sensor types: *temperature, absoluteHumidity, relativeHumidity, airPressure, airFlow* or *vibration*.

  Note: If the specified sensor type does not match the type of the specified environmental sensor, this error message appears: "Specified sensor type 'XXX' does not match the sensor's type (<sensortype>)," where XXX is the specified sensor type, and <sensortype> is the correct sensor type.

- <option> is one of the options: *enable*, *disable* or a numeric value.

Option	Description
enable	Enables the specified threshold for a specific environmental sensor.
disable	Disables the specified threshold for a specific environmental sensor.
A numeric value	Sets a value for the specified threshold of a specific environmental sensor and enables this threshold at the same time.

- <hy_value> is a numeric value that is assigned to the hysteresis for the specified environmental sensor. Refer to ***"To De-assert" and Deassertion Hysteresis*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.
- <as_value> is a number in samples that is assigned to the assertion timeout for the specified environmental sensor. It ranges between 1 and 100. Refer to ***Configuring An Assertion Timeout*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.

Examples

This section illustrates several environmental sensor threshold configuration examples.

Example 1 - Upper Critical Threshold for a Temperature Sensor

The following command sets the Upper Critical threshold of the environmental "temperature" sensor with the ID number 2 to 40 degrees Celsius. It also enables the upper critical threshold if this threshold has not been enabled yet.

```
config:#   sensor externalsensor 2 temperature upperCritical 40
```

The following command sets both the Upper Warning and Lower Warning thresholds for the inlet 1 RMS current.

```
config:#   sensor inlet 1 current upperWarning 20 lowerWarning 12
```

Results:

- The Upper Warning threshold for the inlet 1 RMS current is set to 20A. It also enables the upper warning threshold if this threshold has not been enabled yet.
- The Lower Warning threshold for the inlet 1 RMS current is set to 12A. It also enables the lower warning threshold if this threshold has not been enabled yet.

Example 3 - Upper Thresholds for Overcurrent Protector Sensors

The following command sets both the Upper Critical and Upper Warning thresholds for the 2nd overcurrent protector.

```
config:#   sensor ocp 2 current upperWarning enable upperCritical 16
```

Results:

- The Upper Critical threshold for the 2nd overcurrent protector's RMS current is set to 16A. It also enables the upper critical threshold if this threshold has not been enabled yet.
- The Upper Warning threshold for the 2nd overcurrent protector's RMS current is enabled.

Actuator Configuration Commands

An actuator configuration command begins with *actuator*. You can configure the name and location parameters of an individual actuator.

You can configure various parameters for one actuator at a time. See *Multi-Command Syntax* (on page 122).

- **Change the name:**

```
config:#   actuator <n> name "<name>"
```

- **Set the X coordinate:**

```
config:#   actuator <n> xlabel "<coordinate>"
```

- **Set the Y coordinate:**

```
config:#   actuator <n> ylabel "<coordinate>"
```

- **Set the Z coordinate:**

```
config:#   actuator <n> zlabel "<z_label>"
```

- **Modify the actuator's description:**

```
config:#   actuator <n> description "<description>"
```

Variables:

- <n> is the ID number assigned to the actuator. The ID number can be found using the PDU web interface or CLI. It is an integer starting at 1.
- <name> is a string comprising up to 32 ASCII printable characters. The <name> variable must be enclosed in quotes when it contains spaces.
- <coordinate> is a string comprising up to 24 ASCII printable characters, and it must be enclosed in quotes.
- There are two types of values for the <z_label> variable, depending on the Z coordinate format you set:

Type	Description
Free form	<coordinate> is a string comprising up to 24 ASCII printable characters, and it must be enclosed in quotes.
Rack units	<coordinate> is an integer number in rack units.

Note: To specify the Z coordinate using the rack units, see ***Setting the Z Coordinate Format for Environmental Sensors*** *(on page 46).*

- <description> is a sentence or paragraph comprising up to 64 ASCII printable characters, and it must be enclosed in quotes.

Example - Actuator Naming

The following command assigns the name "Door lock" to the actuator whose ID number is 9.

```
config:#   actuator 9 name "Door lock"
```

Component Reachability Configuration Commands

You can use the CLI to add or delete an device, from the component reachability list, or modify the settings for a monitored device. A component reachability configuration command begins with *componentReachability*.

Adding a Monitored Device

This command adds a new device to the component reachability list.

```
config:#      componentReachability add <IP_host> <enable> <succ_ping> <fail_ping>
              <succ_wait> <fail_wait> <resume> <disable_count>
```

Variables:

- <IP_host> is the IP address or host name of the device that you want to add.
- <enable> is one of the options: *true* or *false*.

Option	Description
true	Enables the ping monitoring feature for the newly added device.
false	Disables the ping monitoring feature for the newly added device.

- <succ_ping> is the number of successful pings for declaring the monitored device "Reachable." Valid range is 0 to 200.
- <fail_ping> is the number of consecutive unsuccessful pings for declaring the monitored device "Unreachable." Valid range is 1 to 100.
- <succ_wait> is the wait time to send the next ping after a successful ping. Valid range is 5 to 600 (seconds).
- <fail_wait> is the wait time to send the next ping after a unsuccessful ping. Valid range is 3 to 600 (seconds).
- <resume> is the wait time before the PDU resumes pinging after declaring the monitored device "Unreachable." Valid range is 5 to 120 (seconds).
- <disable_count> is the number of consecutive "Unreachable" declarations before the PDU disables the ping monitoring feature for the monitored device and returns to the "Waiting for reliable connection" state. Valid range is 1 to 100 or *unlimited*.

Deleting a Monitored Device

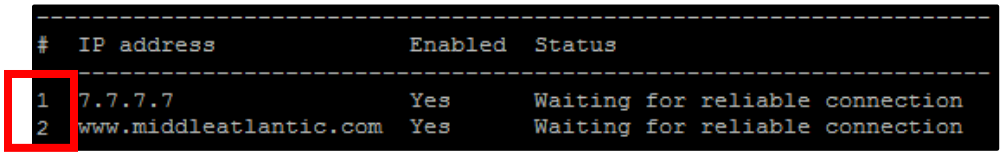
This command removes a monitored device from the component reachability list.

```
config:#   componentReachability delete <n>
```

Variables:

- <n> is a number representing the sequence of the device in the monitored list.

  You can find each device's sequence number using the CLI command of `show componentReachability` as illustrated below.



```
-------------------------------------------------------------------
#  IP address              Enabled  Status
-------------------------------------------------------------------
1  7.7.7.7                 Yes      Waiting for reliable connection
2  www.middleatlantic.com  Yes      Waiting for reliable connection
```

Modifying a Monitored Device's Settings

The command to modify a monitored device's settings begins with *componentReachability modify*.

You can modify various settings for a monitored device at a time. See ***Multi-Command Syntax*** (on page 122).

- **Modify a device's IP address or host name:**

  ```
  config:#   componentReachability modify <n> ipAddress <IP_host>
  ```

- **Enable or disable the ping monitoring feature for the device:**

  ```
  config:#   componentReachability modify <n> pingMonitoringEnabled
             <option>
  ```

- **Modify the number of successful pings for declaring "Reachable":**

  ```
  config:#      componentReachability modify <n> numberOfSuccessfulPingsToEnable <succ_number>
  ```

- **Modify the number of unsuccessful pings for declaring "Unreachable":**

  ```
  config:#   componentReachability modify <n> numberOfUnsuccessfulPingsForFailure
             <fail_number>
  ```

- **Modify the wait time after a successful ping:**

  ```
  config:#     componentReachability modify <n> waitTimeAfterSuccessfulPing <succ_wait>
  ```

- **Modify the wait time after a unsuccessful ping:**

```
config:#      componentReachability modify <n> waitTimeAfterUnsuccessfulPing <fail_wait>
```

- **Modify the wait time before resuming pinging after declaring "Unreachable":**

```
config:#      componentReachability modify <n> waitTimeBeforeResumingPinging <resume>
```

- **Modify the number of consecutive "Unreachable" declarations before disabling the ping monitoring feature:**

```
config:#      componentReachability modify <n> numberOfFailuresToDisable <disable_count>
```

Variables:

- <n> is a number representing the sequence of the device in the component monitoring list.
- <IP_host> is the IP address or host name of the device whose settings you want to modify.
- <option> is one of the options: *true* or *false*.

Option	Description
true	Enables the ping monitoring feature for the monitored device.
false	Disables the ping monitoring feature for the monitored device.

- <succ_number> is the number of successful pings for declaring the monitored device "Reachable." Valid range is 0 to 200.
- <fail_number> is the number of consecutive unsuccessful pings for declaring the monitored device "Unreachable." Valid range is 1 to 100.
- <succ_wait> is the wait time to send the next ping after a successful ping. Valid range is 5 to 600 (seconds).
- <fail_wait> is the wait time to send the next ping after a unsuccessful ping. Valid range is 3 to 600 (seconds).
- <resume> is the wait time before the PDU resumes pinging after declaring the monitored device "Unreachable." Valid range is 5 to 120 (seconds).
- <disable_count> is the number of consecutive "Unreachable" declarations before the PDU disables the ping monitoring feature for the monitored device and returns to the "Waiting for reliable connection" state. Valid range is 1 to 100 or *unlimited*.

Example - Component Settings Changed

The following command modifies several ping monitoring settings for the second server in the component reachability list.

```
config:#      componentReachability modify 2 numberOfSuccessfulPingsToEnable 10
              numberOfUnsuccessfulPingsForFailure 8 waitTimeAfterSuccessfulPing 30
```

Serial Port Configuration Commands

A serial port configuration command begins with *serial*.

Setting the Baud Rates

The following commands set the baud rate (bps) of the serial port labeled CONSOLE on the PDU. Change the baud rate before connecting it to the desired device. If you change the baud rate dynamically after the connection has been made, you must reset the PDU or power cycle the connected device for proper communications.

- **Determine the CONSOLE baud rate:**

```
config:#   serial consoleBaudRate <baud_rate>
```

- **Determine the MODEM baud rate:**

```
config:#   serial modemBaudRate <baud_rate>
```

Variables:

- <baud_rate> is one of the baud rate options: *1200*, *2400*, *4800*, *9600*, *19200*, *38400*, *57600*, *115200*.

Forcing the Device Detection Mode

This command forces the serial port on the PDU to enter a specific device detection mode.

```
config:#   serial deviceDetectionType <mode>
```

Variables:

- <mode> is one of the detection modes: *automatic*, *forceConsole*, *forceAnalogModem,* or *forceGsmModem*.

Option	Description
automatic	The PDU automatically detects the type of the device connected to the serial port. Select this option unless your PDU cannot correctly detect the device type.
forceConsole	The PDU attempts to recognize that the connected device is set for the console mode.

Example

The following command sets the CONSOLE baud rate of the PDU's serial port to 9600 bps.

```
config:#   serial consoleBaudRate 9600
```

Setting the History Buffer Length

This command syntax sets the history buffer length, which determines the amount of history commands that can be retained in the buffer. The default length is 25.

```
config:#   history length <n>
```

Variables:

- <n> is an integer number between 1 and 250.

Multi-Command Syntax

To shorten the configuration time, you can combine various configuration commands in one command to perform all of them at a time. All combined commands must belong to the same configuration type, such as commands prefixed with *network*, *user modify*, *sensor externalsensor* and so on.

A multi-command syntax looks like this:

```
<configuration type> <setting 1> <value 1> <setting 2> <value 2> <setting 3> <value 3> ...
```

Example 1 - Combination of IP, Subnet Mask and Gateway Parameters

The following multi-command syntax configures IPv4 address, subnet mask and gateway for the network connectivity simultaneously.

```
config:#      network ipv4 ipAddress 192.168.84.225 subnetMask 255.255.255.0 gateway
              192.168.84.0
```

Results:

- The IP address is set to 192.168.84.225.
- The subnet mask is set to 255.255.255.0.
- The gateway is set to 192.168.84.0.

Example 2 - Combination of Upper Critical and Upper Warning Settings

The following multi-command syntax simultaneously configures Upper Critical and Upper Warning thresholds for the RMS current of the 2nd overcurrent protector.

```
config:#   sensor ocp 2 current upperCritical disable upperWarning 15
```

Results:

- The Upper Critical threshold of the 2nd overcurrent protector's RMS current is disabled.
- The Upper Warning threshold of the 2nd overcurrent protector's RMS current is set to 15A and enabled at the same time.

Example 3 - Combination of SSID and PSK Parameters

This multi-command syntax configures both SSID and PSK parameters simultaneously for the wireless feature.

```
config:#   network wireless SSID myssid PSK encryp_key
```

Results:

- The SSID value is set to myssid.
- The PSK value is set to encryp_key.

Example 4 - Combination of Upper Critical, Upper Warning and Lower Warning Settings

The following multi-command syntax configures Upper Critical, Upper Warning and Lower Warning thresholds for the outlet 5 RMS current simultaneously.

```
config:#     sensor outlet 5 current upperCritical disable upperWarning enable lowerWarning
             1.0
```

Results:

- The Upper Critical threshold of outlet 5 RMS current is disabled.
- The Upper Warning threshold of outlet 5 RMS current is enabled.
- The Lower Warning threshold of outlet 5 RMS current is set to 1.0A and enabled at the same time.

Load Shedding Configuration Commands

A load shedding configuration command begins with *loadshedding*.

Unlike other CLI configuration commands, the load shedding configuration command is performed in the *administrator mode* rather than the configuration mode. See ***Different CLI Modes and Prompts*** (on page 20).

Enabling or Disabling Load Shedding

This command determines whether to enter or exit from the load shedding mode.

```
#          loadshedding <option>
```

After performing the above command, the PDU prompts you to confirm the operation. Press y to confirm or n to abort the operation.

To skip the confirmation step, you can add the "/y" parameter to the end of the command so that the operation is executed immediately.

```
#          loadshedding <option> /y
```

Variables:

- <option> is one of the options: *enable* or *disable*.

Option	Description
start	Enter the load shedding mode.
stop	Quit the load shedding mode.

Example

The following command puts the PDU into load shedding mode.

```
config:#   loadshedding start
```

Power Control Operations

Outlets on the PDU can be turned on or off or power cycled through the CLI.

Besides, you can cancel the power-on process while the PDU is powering on ALL outlets.

You must perform this operation in the *administrator mode*. See ***Different CLI Modes and Prompts*** (on page 20).

Turning On the Outlet(s)

This command turns on one or multiple outlets.

```
#        power outlets <numbers> on
```

To quicken the operation, you can add the parameter "/y" to the end of the command, which confirms the operation.

```
#        power outlets <numbers> on /y
```

Variables:

- <numbers> is one of the options: *all*, an outlet number, a list or a range of outlets.

Option	Description
all	Switches ON all outlets.

Option	Description
A specific outlet number	Switches ON the specified outlet.
A comma- separated list of outlets	Switches ON multiple, inconsecutive or consecutive outlets. For example, to specify 7 outlets -- 2, 4, 9, 11, 12, 13 and 15, type `outlets 2,4,9,11-13,15`.
A range of outlets with an en dash in between	Switches ON multiple, consecutive outlets. For example, to specify 6 consecutive outlets -- 3, 4, 5, 6, 7, 8, type `outlets 3-8`.

If you entered the command without "/y", a message appears, prompting you to confirm the operation. Then:

- Type y to confirm the operation, OR
- Type n to abort the operation

Turning Off the Outlet(s)

This command turns off one or multiple outlets.

```
#       power outlets <numbers> off
```

To quicken the operation, you can add the parameter "/y" to the end of the command, which confirms the operation.

```
#       power outlets <numbers> off /y
```

Variables:

- <numbers> is one of the options: *all*, an outlet number, a list or a range of outlets.

Option	Description
all	Switches OFF all outlets.
A specific outlet number	Switches OFF the specified outlet.
A comma- separated list of outlets	Switches OFF multiple, inconsecutive or consecutive outlets. For example, to specify 7 outlets -- 2, 4, 9, 11, 12, 13 and 15, type `outlets 2,4,9,11-13,15`.
A range of outlets with an en dash in between	Switches OFF multiple, consecutive outlets. For example, to specify 6 consecutive outlets -- 3, 4, 5, 6, 7, 8, type `outlets 3-8`.

If you entered the command without "/y", a message appears, prompting you to confirm the operation. Then:

- Type y to confirm the operation, OR
- Type n to abort the operation

Power Cycling the Outlet(s)

This command power cycles one or multiple outlets.

```
#       power outlets <numbers> cycle
```

To quicken the operation, you can add the parameter "/y" to the end of the command, which confirms the operation.

```
#       power outlets <numbers> cycle /y
```

Variables:

- <numbers> is one of the options: *all*, an outlet number, a list or a range of outlets.

Option	Description
all	Power cycles all outlets.
A specific outlet number	Power cycles the specified outlet.
A comma- separated list of outlets	Power cycles multiple, inconsecutive or consecutive outlets. For example, to specify 7 outlets -- 2, 4, 9, 11, 12, 13 and 15, type `outlets 2,4,9,11-13,15`.
A range of outlets with an en dash in between	Power cycles multiple, consecutive outlets. For example, to specify 6 consecutive outlets -- 3, 4, 5, 6, 7, 8, type `outlets 3-8`.

If you entered the command without "/y", a message appears, prompting you to confirm the operation. Then:

- Type y to confirm the operation, OR
- Type n to abort the operation

Canceling the Power-On Process

After issuing the command to power on ALL outlets, you can use the following command to stop the power-on process.

```
#       power cancelSequence
```

To quicken the operation, you can add the parameter "/y" to the end of the command, which confirms the operation.

```
#       power cancelSequence /y
```

Example - Power Cycling Specific Outlets

The following command power cycles these outlets: 2, 6, 7, 8, 10, 13, 14, 15 and 16.

```
#     power outlets 2,6-8,10,13-16 cycle
```

Actuator Control Operations

An actuator, which is connected to a dry contact signal channel of a DX sensor, can control a mechanism or system. You can switch on or off that mechanism or system through the actuator control command in the CLI.

Perform these commands in the administrator or user mode. See *Different CLI Modes and Prompts* (on page 20).

Switching On an Actuator

This command syntax turns on one actuator.

```
#       control actuator <n> on
```

To quicken the operation, you can add the parameter "/y" to the end of the command, which confirms the operation.

```
#       control actuator <n> on /y
```

Variables:

- <n> is an actuator's ID number.

  The ID number is available in the PDU web interface or using the show command in the CLI. It is an integer between 1 and 32.

If you entered the command without "/y", a message appears, prompting you to confirm the operation. Then:

- Type y to confirm the operation, OR
- Type n to abort the operation

Switching Off an Actuator

This command syntax turns off one actuator.

```
#       control actuator <n> off
```

To quicken the operation, you can add the parameter "/y" to the end of the command, which confirms the operation.

```
#       control actuator <n> off /y
```

Variables:

- <n> is an actuator's ID number.

  The ID number is available in the PDU web interface or using the show command in the CLI. It is an integer between 1 and 32.

If you entered the command without "/y", a message appears, prompting you to confirm the operation. Then:

- Type y to confirm the operation, OR
- Type n to abort the operation

Example - Turning On a Specific Actuator

The following command turns on the actuator whose ID number is 8.

```
#     control actuator 8 on
```

Unblocking a User

If any user is blocked from accessing the PDU, you can unblock them at the local console.

To unblock a user:

1. Log in to the CLI interface using any terminal program via a local connection. See *With HyperTerminal* (on page 18).
2. When the Username prompt appears, type `unblock` and press Enter.

```
Username: unblock
```

3. When the "Username to unblock" prompt appears, type the name of the blocked user and press Enter.

```
Username to unblock:
```

4. A message appears, indicating that the specified user was unblocked successfully.

Resetting the PDU

You can reset the PDU to factory defaults or simply restart it using the CLI commands.

Restarting the PDU

This command restarts the PDU. This is not a factory default reset.

To restart the PDU:

1. Ensure you have entered administrator mode and the # prompt is displayed.

2. Type either of the following commands to restart the PDU.

```
#      reset unit
```

-- OR --

```
#      reset unit /y
```

3. If you entered the command without "/y" in Step 2, a message appears prompting you to confirm the operation. Type y to confirm the reset.
4. Wait until the Username prompt appears, indicating the reset is complete.

Note: If you are performing this command over a USB connection, re-connect the USB cable after the reset is completed, or the CLI communications are lost.

Resetting Active Energy Readings

You can reset either one active energy sensor or all active energy sensors at a time to restart the energy accumulation process.

Only users with the "Admin" role assigned can reset active energy readings.

- **To reset all active energy readings of the PDU:**

```
#      reset activeEnergy pdu
```

-- OR --

```
#      reset activeEnergy pdu /y
```

- **To reset one inlet's active energy readings:**

```
#      reset activeEnergy inlet <n>
```

-- OR --

```
#      reset activeEnergy inlet <n> /y
```

- **To reset one outlet's active energy readings:**

```
#      reset activeEnergy outlet <outlet_n>
```

-- OR --

```
#      reset activeEnergy outlet <outlet_n> /y
```

If you entered the command without "/y", a message appears prompting you to confirm the operation. Type y to confirm the reset or n to abort it.

Variables:

- <n> is the inlet number.
- <outlet_n> is an outlet number.

Resetting to Factory Defaults

The following commands restore all settings of the PDU to factory defaults.

▶ **To reset PDU settings after login, use either command:**

```
#     reset factorydefaults
```

-- OR --

```
#     reset factorydefaults /y
```

▶ **To reset PDU settings before login:**

```
Username:   factorydefaults
```

For more information refer to ***Using the CLI Command*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.

Network Troubleshooting

The PDU provides 4 diagnostic commands for troubleshooting network problems: *nslookup*, *netstat*, *ping*, and *traceroute*. The diagnostic commands function as corresponding Linux commands and can get corresponding Linux outputs.

Entering Diagnostic Mode

Diagnostic commands function in the diagnostic mode only.

▶ **To enter the diagnostic mode:**

1. Enter either of the following modes:
   - Administrator mode: The # prompt is displayed.
   - User mode: The > prompt is displayed.
2. Type `diag` and press Enter. The diag# or diag> prompt appears, indicating that you have entered the diagnostic mode.
3. Now you can type any diagnostic commands for troubleshooting.

Quitting Diagnostic Mode

- **To quit the diagnostic mode, use this command:**

```
diag>       exit
```

The # or > prompt appears after pressing Enter, indicating that you have entered the administrator or user mode. See *Different CLI Modes and Prompts* (on page 20).

Diagnostic Commands

The diagnostic command syntax varies from command to command.

Querying DNS Servers

This command syntax queries Internet domain name server (DNS) information of a network host.

```
diag>       nslookup <host>
```

Variables:

- <host> is the name or IP address of the host whose DNS information you want to query.

Showing Network Connections

This command syntax displays network connections and/or status of ports.

```
diag>       netstat <option>
```

Variables:

- <option> is one of the options: *ports* or *connections*.

Option	Description
ports	Shows TCP/UDP ports.
connections	Shows network connections.

Testing the Network Connectivity

This ping command sends the ICMP ECHO_REQUEST message to a network host for checking its network connectivity. If the output shows the host is responding properly, the network connectivity is good. If not, either the host is shut down or it is not being properly connected to the network.

```
diag>       ping <host>
```

Variables:

- <host> is the host name or IP address whose networking connectivity you want to check.

Options:

- You can include any or all of additional options listed below in the ping command.

Options	Description
count <number1>	Determines the number of messages to be sent. <number1> is an integer number between 1 and 100.
size <number2>	Determines the packet size. <number2> is an integer number in bytes between 1 and 65468.
timeout <number3>	Determines the waiting period before timeout. <number3> is an integer number in seconds ranging from 1 to 600.

The command looks like the following when it includes all options:

```
diag>      ping <host> count <number1> size <number2> timeout <number3>
```

Tracing the Route

This command syntax traces the network route between your PDU and a network host.

```
diag>      traceroute <host>
```

Variables:

- <host> is the name or IP address of the host you want to trace.

Example - Ping Command

The following command checks the network connectivity of the host 192.168.84.222 by sending the ICMP ECHO_REQUEST message to the host for 5 times.

```
diag>      ping 192.168.84.222 count 5
```

Retrieving Previous Commands

If you would like to retrieve any command that was previously typed in the same connection session, press the Up arrow (↑) on the keyboard until the desired command is displayed.

Automatically Completing a Command

A CLI command always consists of several words. You can easily enter a command by typing first word(s) or letter(s) and then pressing Tab or Ctrl+i instead of typing the whole command word by word.

To have a command completed automatically:

1. Type initial letters or words of the desired command. Make sure the letters or words you typed are unique so that the CLI can identify the command you want.
2. Press Tab or Ctrl+i until the complete command appears.

Example 1:

Type the first word and the first letter of the second word of the "`reset factorydefaults`" command, that is, `reset f`. Then press Tab or Ctrl+i to complete the second word.

Example 2:

Type the first word and initial letters of the second word of the "`security enforceHttpsForWebAccess`" command, that is, `security enf`. Then press Tab or Ctrl+i to complete the second word.

Logging out of the CLI

After completing your tasks using the CLI, always log out of the CLI to prevent others from accessing the CLI.

To log out of the CLI:

1. Ensure you have entered administrator mode and the # prompt is displayed.
2. Type `exit` and press Enter.

Chapter 5: Bulk Configuration Methods

If you have to set up multiple PDUs, you can use one of the following configuration methods to save time.

Use a bulk configuration file:

- Requirement: All PDUs to configure are of the same model and firmware.
- Procedure: First finish configuring one PDU. Then save the bulk configuration file from it and copy this file to all of the other PDUs.

  Refer to ***Bulk Configuration*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.

Use a TFTP server:

- Requirement: DHCP is enabled in your network and a TFTP server is available.
- Procedure: Prepare special configuration files, which must include *fwupdate.cfg*, and copy them to the root directory of the TFTP server. Re-boot all PDUs after connecting them to the network.

  See ***Bulk Configuration or Firmware Upgrade via DHCP/TFTP*** (on page 134).

Use a USB flash drive:

- Requirement: A FAT32- or superfloppy-formatted USB flash drive containing special configuration files is required.
- Procedure: Plug the USB drive into the PDU. When a happy smiley is shown on the front panel display, press and hold one of the control buttons on the front panel until the display turns blank.
- See ***Configuration or Firmware Upgrade with a USB Drive*** (on page 136).

Bulk Configuration or Firmware Upgrade via DHCP/TFTP

If a TFTP server is available, you can use it and appropriate configuration files to perform any or all of the following tasks for a large number of PDUs in the same network.

- Initial deployment
- Configuration changes
- Firmware upgrade
- Downloading diagnostic data

This feature is extremely useful if you have hundreds or even thousands of PDUs to configure or upgrade.

Warning: The feature of bulk configuration or firmware upgrade via DHCP/TFTP only works on standalone PDUs directly connected to the network.

*Tip: For the alternative, see **Configuration or Firmware Upgrade with a USB Drive** (on page 136).*

Bulk Configuration/Upgrade Procedure

The DHCP/TFTP feature is supported, so make sure that all PDUs which you want to configure or upgrade are running the latest firmware version.

Steps of using DHCP/TFTP for bulk configuration/upgrade:

1. Create configuration files specific to your PDU models and firmware versions. See ***Configuration Files*** (on page 137) or contact Technical Support to properly prepare some or all of the following files:
   - *fwupdate.cfg (always required)*
   - *config.txt*
   - *devices.csv*

   Note: Supported syntax of "fwupdate.cfg" and "config.txt" may vary based on different firmware versions. If you have existing configuration files, it is suggested to double check with Technical Support for the correctness of these files prior to using this feature.
2. Configure your TFTP server properly. See ***TFTP Requirements*** (on page 135).
3. Copy ALL required configuration files into the TFTP root directory. If the tasks you will perform include firmware upgrade, an appropriate firmware binary file is also required.
4. Properly configure your DHCP server so that it refers to the file "fwupdate.cfg" on the TFTP server for your PDU.
5. Make sure all of the desired PDUs use DHCP as the IP configuration method and have been *directly* connected to the network.
6. Re-boot the PDUs. The DHCP server will execute the commands in the "fwupdate.cfg" file on the TFTP server to configure or upgrade the PDUs supporting DHCP in the same network.

   DHCP will execute the "fwupdate.cfg" commands once for IPv4 and once for IPv6 respectively if both IPv4 and IPv6 settings are configured properly in DHCP.

TFTP Requirements

To perform bulk configuration or firmware upgrade successfully, your TFTP server must meet the following requirements:

- The server is able to work with both IPv4 and IPv6.

In Linux, remove any IPv4 or IPv6 flags from */etc/xinetd.d/tftp*.

Note: DHCP will execute the "fwupdate.cfg" commands once for IPv4 and once for IPv6 respectively if both IPv4 and IPv6 settings are configured properly in DHCP.

- All required configuration files are available in the TFTP root directory. See ***Bulk Configuration/Upgrade Procedure*** (on page 135).

If you are going to upload any PDU diagnostic file or create a log file in the TFTP server, the first of the following requirements is also required.

- The TFTP server supports the write operation, including file creation and upload.

  In Linux, provide the option "-c" for write support.
- Required for uploading the diagnostic file only - the timeout for file upload is set to one minute or larger.

Configuration or Firmware Upgrade with a USB Drive

You can accomplish part or all of the following tasks simultaneously by plugging a USB flash drive which contains one or several special configuration files into the PDU.

- Configuration changes
- Firmware upgrade
- Downloading diagnostic data

*Tip: You can also accomplish the same tasks via the TFTP server in a DHCP network. See **Bulk Configuration or Firmware Upgrade via DHCP/TFTP** (on page 134).*

Device Configuration/Upgrade Procedure

You can use one USB drive to configure or upgrade multiple PDUs one by one as long it contains valid configuration files.

▶ **To use a USB drive to configure the PDU or upgrade firmware:**

1. Verify that both the USB drive and your PDU meet the requirements. See ***System and USB Requirements*** (on page 137).
2. Prepare required configuration files. See ***Configuration Files*** (on page 137).
3. Copy required configuration files to the root directory of the USB drive.
   - For firmware upgrade, an appropriate firmware binary file is also required.
4. Plug the USB drive into the USB-A port of the PDU.
5. Wait for several seconds until the PDU resumes normal operation, indicated by the normal message of the display.

If nothing is shown on the display and no task is performed after plugging the USB drive, check the log file in the USB drive.

System and USB Requirements

You must satisfy ALL of the following requirements prior to using a USB flash drive to perform device configuration and/or firmware upgrade.

PDU system requirements:

- There is at least one USB-A port available on your device.
- Your PDU must be running the latest version of the firmware.

  Note that the PDU interpreted the USB drive's contents using the firmware which was running when plugging the USB drive, not the new firmware after firmware upgrade.

USB drive requirements:

- The drive contains either a single partition formatted as a Windows FAT32 filesystem, or NO partition tables (that is, a superfloppy-formatted drive).
- The drive contains a configuration file called *fwupdate.cfg* in its root directory. See ***fwupdate.cfg*** (on page 137).

Configuration Files

There are three types of configuration files.

- **fwupdate.cfg:**

  This file MUST be always present for performing configuration or firmware upgrade tasks. See ***fwupdate.cfg*** (on page 137).

- **config.txt:**

  This file is used for configuring device settings. See ***config.txt*** (on page 140).

- **devices.csv:**

  This file is required only when there are device-specific settings to configure for multiple PDUs. See ***devices.csv*** (on page 142).

A Mass Deployment Utility is provided, which helps you to quickly generate all configuration files for your PDU. See ***Creating Configuration Files via Mass Deployment Utility*** (on page 143).

fwupdate.cfg

The configuration file, *fwupdate.cfg*, is an ASCII text file containing key-value pairs, one per line.

Each value in the file must be separated by an equal sign (=), without any surrounding spaces. Keys are not case sensitive.

Illustration:

```
user=admin
password=raritan
logfile=log.txt
config=config.txt
device_list=devices.csv
```

This section only explains common options in the file.

user

- A required option.
- Specify the name of a user account with Administrator Privileges.
- For a PDU with factory default configuration, set this option to `admin`.

password

- A required option.
- Specify the password of the specified admin user.
- For a PDU with factory default configuration, set this option to `admin`.

logfile

- Specify the name of a text file where the PDU will append the log messages when interpreting the USB drive contents.
- If the specified file does not exist in the USB drive, it will be automatically created.
- If this option is not set, no log messages are recorded. The disadvantage is that no feedback is available if the PDU detects a problem with the USB drive contents.

firmware

- Specify the name of a firmware binary file used to upgrade your PDU.
- The specified firmware file must be compatible with your PDU and have an official Middle Atlantic Products signature.
- If the specified firmware file is the same as the current firmware version of your PDU, no firmware upgrade is performed unless you have set the option "force_update" to `true`.

force_update

- If this option is set to `true`, the firmware upgrade is always performed even though your PDU is running the same firmware version as the specified firmware file.
- This option CANNOT break other constraints like the minimum downgrade version.

config

- Specify the name of the configuration file containing device settings.
- The suggested filename is *config.txt*. See ***config.txt*** (on page 140).

device_list

- Specify the name of the configuration file listing all PDUs to configure and their device-specific settings.
- This file is required if any macros are used in the device configuration file "config.txt."
- The suggested filename is *devices.csv*. See ***devices.csv*** (on page 142).

match

- Specify a match condition for identifying a line or a PDU in the device configuration file "devices.csv."

  The option's value comprises one word and one number as explained below:

  - The word prior to the colon is an identification property, which is either `serial` for serial number or `mac` for MAC address.
  - The number following the colon indicates a column in the *devices.csv* file.

  For example, `mac:7` instructs the PDU to search for the MAC address in the 7th column of the "devices.csv" file.
- The default value is `serial:1`, making the PDU search for its serial number in the first column.
- This option is used only if the "device_list" option has been set.

collect_diag

- If this option is set to `true`, the diagnostic data of the PDU is downloaded to the USB drive.
- The filename of the diagnostic data written into the USB drive varies, depending on the PDU firmware version. Consider the following example:
  - Filename prior to version 3.0.0: *diag_<unit-serial>.zip*, where <unit-serial> is the serial number of the PDU.
  - Filename as of version 3.0.0: *diag_<unit-serial>.tgz*
- The PDU writes the diagnostic data to the USB drive.

factory_reset

- If this option is set to `true`, the PDU will be reset to factory defaults.
- If the device configuration will be updated at the same time, the factory reset will be executed before updating the device configuration.

bulk_config_restore

- Supported as of release 3.1.0.
- Specify the name of the bulk configuration file used to configure or restore the PDU.

Note: For instructions on generating a bulk configuration file, refer to ***Bulk Configuration*** *in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.*

- Additional configuration keys set via the *config.txt* file will be applied after performing the bulk restore operation.
- This option CANNOT be used with the option "full_config_restore."
- If a firmware upgrade will be performed at the same time, you must generate the bulk configuration file based on the NEW firmware version instead of the current firmware version.

full_config_restore

- Specify the name of the full configuration backup file used to restore the PDU.

Note: For instructions on generating the full configuration backup file, refer to ***Backup and Restore of Device Settings*** *in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.*

- Additional configuration keys set via the *config.txt* file will be applied after performing the configuration restore operation.
- This option CANNOT be used with the option "bulk_config_restore."
- If a firmware upgrade will be performed at the same time, you must generate the full configuration backup file based on the NEW firmware version instead of the current firmware version.

config.txt

To perform device configuration using a USB drive, you must:

- Copy the device configuration file "config.txt" to the root directory of the USB drive.
- Reference the "config.txt" file in the *config* option of the "fwupdate.cfg" file. See ***fwupdate.cfg*** (on page 137).

The file, *config.txt*, is a text file containing a number of configuration keys and values to configure or update.

This section only introduces the device configuration file in brief, and does not document all configuration keys, which vary according to the firmware version and your PDU model.

You can use the Mass Deployment Utility to create this file yourself, or contact technical support to get a device configuration file specific to your PDU model and firmware version.

Tip: As of release 3.2.20, you can choose to encrypt important data in the "config.txt" file so that people cannot easily recognize it, such as the SNMP write community string. See ***Data Encryption in 'config.txt'*** *(on page 144)*.

Regular configuration key syntax:

- Each configuration key and value pair is in a single line as shown below:
  `key=value`

Note: Each value in the file must be separated by an equal sign (=), without any surrounding spaces.

- As of release 3.1.0, multi-line values are supported by using the *Here Document Syntax* with a user-chosen delimiter.

  The following illustration declares a value in two lines. You can replace the delimiter `EOF` with other delimiter strings.

```
key<<EOF
value line 1
value line 2
EOF
```

Note: The line break before the closing EOF is not part of the value. If a line break is required in the value, insert an additional empty line before the closing EOF.

Special configuration keys:

There are 3 special configuration keys that are prefixed with `magic:`.

- A special key that sets a user account's password without knowing the firmware's internal encryption/hashing algorithms is implemented as of release 2.2.13.

  Example:
  `magic:users[1].cleartext_password=joshua`
- Two special keys that set the SNMPv3 passphrases without knowing the firmware's internal encryption/hashing algorithms are implemented as of release 2.4.0.

  Examples:
  `magic:users[1].snmp_v3.auth_phrase=swordfish`
  `magic:users[1].snmp_v3.priv_phrase=opensesame`

To configure device-specific settings:

1. Make sure the device list configuration file "devices.csv" is available in the USB drive. See ***devices.csv*** (on page 142).
2. In the "config.txt" file, refer each device-specific configuration key to a specific column in the "devices.csv" file. The syntax is: `${column}`, where "column" is a column number.

   Examples:
   `network.interfaces[eth0].ipaddr=${2}`
   `pdu.name=${16}`

To rename the admin user:

As of release 3.1.0, you can rename the admin user by adding the following configuration key:

```
users[0].name=new admin name
```

Example:

```
users[0].name=May
```

devices.csv

If there are device-specific settings to configure, you must create a device list configuration file - *devices.csv*, to store unique data of each PDU.

This file must be:

- An excel file in the CSV format.
- Copied to the root directory.
- Referenced in the *device_list* option of the "fwupdate.cfg" file. See ***fwupdate.cfg*** (on page 137).

Every PDU identifies its entry in the "devicelist.csv" file by comparing its serial number or MAC address to one of the columns in the file.

Determine the column to identify PDUs:

- By default, a PDU searches for its serial number in the 1st column.
- To override the default, set the *match* option in the "fwupdate.cfg" file to a different column.

Syntax:

- Prior to release 3.1.0, only single-line values containing NO commas are supported. A comma is considered a field delimiter.

  For example:

  ```
  Value-1,Value-2,Value-3
  ```

- As of release 3.1.0, values containing commas, line breaks or double quotes are all supported. The commas and line breaks to be included in the values must be enclosed in double quotes. Every double quote to be included in the value must be escaped with another double quote.

For example:

```
Value-1,"Value-2,with,three,commas",Value-3

Value-1,"Value-2,""with""three""double-quotes",Value-3

Value-1,"Value-2
with a line break", Value-3
```

Creating Configuration Files via Mass Deployment Utility

The Mass Deployment Utility is an Excel file that lets you fill in basic information required for the three configuration files, such as the admin account and password.

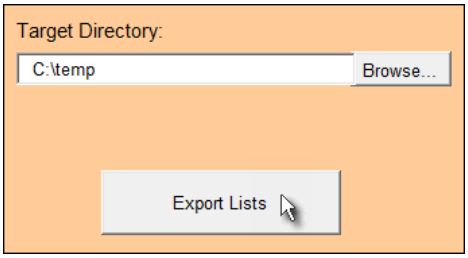
After entering required information, you can generate all configuration files with only one click, including *fwupdate.cfg*, *config.txt* and *devices.csv*.

To use the Mass Deployment Utility:

1. Download the Mass Deployment Utility from the website.
   - The utility is named *mass_deployment-xxx* (where xxx is the firmware version number).
   - It is available at www.middleatlantic.com/resources/power-downloads.aspx.
2. Launch Excel to open this utility.

   Note: Other office suites, such as OpenOffice and LibreOffice, are not supported.
3. Read the instructions in the 1st worksheet of the utility, and make sure Microsoft Excel's security level has been set to Medium or the equivalent for executing unsigned macros of this utility.
4. Enter information in the 2nd and 3rd worksheets.
   - The 2nd worksheet contains information required for *fwupdate.cfg* and *config.txt*.
   - The 3rd worksheet contains device-specific information for *devices.csv*.
5. Return to the 2nd worksheet to execute the export macro.
   a. In the Target Directory field, specify the folder where to generate the configuration files. For example, you can specify the root directory of a connected USB drive.

b. Click Export Lists to generate configuration files.



6. Verify that at least 3 configuration files are created - *fwupdate.cfg*, *config.txt* and *devices.csv*. You are ready to configure or upgrade any PDU with these files. See ***Configuration or Firmware Upgrade with a USB Drive*** (on page 136).

Data Encryption in 'config.txt'

Encryption for any settings in the file "config.txt" is supported as of release 1.0.0.

When intending to prevent people from identifying the values of any settings, you can encrypt them. Encrypted data still can be properly interpreted and performed by any PDU running the latest firmware version.

Data encryption procedure:

1. Open the "config.txt" file to determine which setting(s) to encrypt.
   - If an appropriate "config.txt" is not created yet, see ***Creating Configuration Files via Mass Deployment Utility*** (on page 143).
2. Launch a terminal to log in to the CLI of any PDU running the latest firmware version. See ***Logging in to the CLI*** (on page 18).
3. Type the encryption command and the value of the setting you want to encrypt.
   - The value *cannot* contain any double quotes (") or backslashes (-).
   - If the value contains spaces, it must be enclosed in double quotes.

```
# config encrypt <value>
```

```
-- OR --
```

```
# config encrypt "<value with spaces>"
```

4. Press Enter. The CLI generates and displays the encrypted form of the typed value.
5. Go to the "config.txt" file and replace the chosen value with the encrypted one by typing or copying the encrypted value from the CLI.
6. Add the text "encrypted:" to the beginning of the encrypted setting.
7. Repeat steps 3 to 6 for additional settings you intend to encrypt.

8. Save the changes made to the "config.txt" file. Now you can use this file to configure any PDU running the latest firmware version. See *Configuration or Firmware Upgrade with a USB Drive* (on page 136).

Illustration:

In this example, we will encrypt the word "private", which is the value of the SNMP write community in the "config.txt" file.

```
snmp.write_community=private
```

1. In the CLI, type the following command to encrypt "private."

```
# config encrypt private
```

2. The CLI generates and shows the encrypted form of "private."

```
ZTtnYcvQUw==
```

3. In the "config.txt" file, make the following changes to the SNMP write community setting.
   a. Replace the word "private" with the encrypted value that CLI shows.

```
snmp.write_community=ZTtnYcvQUw==
```

   b. Add "encrypted:" to the beginning of that setting.

```
encrypted:snmp.write_community=ZTtnYcvQUw==
```

Firmware Upgrade via USB

Firmware files are available at www.middleatlantic.com/resources/power-downloads.aspx.

Note that if the firmware file used for firmware upgrade is the same as the firmware version running on the PDU, no firmware upgrade will be performed unless you have set the *force_update* option to true in the "fwupdate.cfg" file. See *fwupdate.cfg* (on page 137).

To use a USB drive to upgrade the PDU:

1. Copy the configuration file "fwupdate.cfg" and an appropriate firmware file to the root directory of the USB drive.
2. Reference the firmware file in the *image* option of the "fwupdate.cfg" file.

3. Plug the USB drive into the USB-A port on the PDU.
4. The PDU performs the firmware upgrade.
   - The front panel display shows the firmware upgrade progress.

   Tip: You can remove the USB drive and plug it into another PDU for firmware upgrade when the firmware upgrade message displays.
5. It may take one to five minutes to complete the firmware upgrade, depending on your product.
6. When the firmware upgrade finishes, the front panel display indicates the firmware upgrade result.

Chapter 6: Using Secure Copy (SCP) Commands

You can perform a Secure Copy (SCP) command to update the PDU firmware, do bulk configuration, or back up and restore the configuration.

Firmware Update via SCP

During all PDU firmware updates, all user management operations are suspended and all login attempts fail during the process. For details, refer to ***Updating the PDU Firmware*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.

Warning: Do NOT perform the firmware upgrade over a wireless network connection.

To update the firmware via SCP:

1. Type the following SCP command and press Enter.
   `scp <firmware file> <user name>@<device ip>:/fwupdate`

   - *<firmware file>* is the PDU firmware's filename. If the firmware file is not in the current directory, you must include the path in the filename.
   - *<user name>* is the "admin" or any user profile with the Firmware Update permission.
   - *<device ip>* is the IP address of the PDU that you want to update.
2. When the system prompts you to enter the password for the specified user profile, type it and press Enter.
3. The system transmits the specified firmware file to the PDU, and shows the transmission speed and percentage.
4. When the transmission is complete, it shows the following message, indicating that the PDU starts to update its firmware now. Wait until the upgrade completes.
   `Starting firmware update. The connection will be closed now.`

SCP example:

`scp  pdu-rlnk-030000-41270.bin  admin@192.168.87.50:/fwupdate`

Windows PSCP command:

PSCP in Windows works in a similar way to the SCP.

- `pscp <firmware file> <user name>@<device ip>:/fwupdate`

Bulk Configuration via SCP

Like performing bulk configuration via the web interface, there are two steps with the bulk configuration using the SCP commands:

1. Save a configuration from a source PDU.
2. Copy the configuration file to one or multiple destination PDUs.

For detailed information on the bulk configuration requirements, refer to ***Bulk Configuration*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.

To save the configuration via SCP:

1. Type the following SCP command and press Enter.
   `scp <user name>@<device ip>:/bulk_config.xml`
   - *<user name>* is the "admin" or any user profile with the administrator privileges.
   - *<device ip>* is the IP address of the PDU whose configuration you want to save.
2. Type the user password when prompted.
3. The system saves the configuration from the PDU to a file named "bulk_config.xml."

To copy the configuration via SCP:

1. Type the following SCP command and press Enter.
   `scp bulk_config.xml <user name>@<device ip>:/bulk_restore`
   - *<user name>* is the "admin" or any user profile with the administrator privileges.
   - *<device ip>* is the IP address of the PDU whose configuration you want to copy.
2. Type the user password when prompted.
3. The system copies the configuration included in the file "bulk_config.xml" to another PDU, and displays the following message.
   `Starting restore operation. The connection will be closed now.`

SCP examples:

- Save operation:
  `scp  admin@192.168.87.50:/bulk_config.xml`
- Copy operation:
  `scp  bulk_config.xml admin@192.168.87.47:/bulk_restore`

Windows PSCP commands:

PSCP in Windows works in a similar way to the SCP.

- Save operation:
  `pscp <user name>@<device ip>:/bulk_config.xml`
- Copy operation:
  `pscp bulk_config.xml <user name>@<device ip>:/bulk_restore`

Backup and Restore via SCP

To back up ALL settings of a PDU, including device-specific settings, you should perform the backup operation instead of the bulk configuration.

You can restore all settings to previous ones after a backup file is available.

To back up the settings via SCP:

1. Type the following SCP command and press Enter.
   `scp <user name>@<device ip>:/backup_settings.xml`
   - *<user name>* is the "admin" or any user profile with the administrator privileges.
   - *<device ip>* is the IP address of the PDU whose settings you want to back up.
2. Type the user password when prompted.
3. The system saves the settings from the PDU to a file named "backup_settings.xml."

To restore the settings via SCP:

1. Type the following SCP command and press Enter.
   `scp backup_settings.xml <user name>@<device ip>:/settings_restore`
   - *<user name>* is the "admin" or any user profile with the administrator privileges.
   - *<device ip>* is the IP address of the PDU whose settings you want to restore.
2. Type the user password when prompted.
3. The system copies the configuration included in the file "backup_settings.xml" to the PDU, and displays the following message.
   `Starting restore operation. The connection will be closed now.`

SCP examples:

- Backup operation:
  `scp  admin@192.168.87.50:/backup_settings.xml`

- Restoration operation:
  ```
  scp  backup_settings.xml admin@192.168.87.50:/settings_restore
  ```

Windows PSCP commands:

PSCP in Windows works in a similar way to the SCP.

- Backup operation:
  ```
  pscp <user name>@<device ip>:/backup_settings.xml
  ```
- Restoration operation:
  ```
  pscp backup_settings.xml <user name>@<device ip>:/settings_restore
  ```

Downloading Diagnostic Data via SCP

You can download the diagnostic data via SCP.

To download the diagnostic data via SCP:

1. Type the following SCP command and press Enter.
   ```
   scp <user name>@<device ip>:/diag-data.tgz
   ```
   - *<user name>* is the "admin" or any user profile with the Administrator or "Unrestricted View Privileges" privileges.
   - *<device ip>* is the IP address of the PDU whose diagnostic data you want to download.
2. Type the password when the system prompts you to type it.
3. The system saves the diagnostic data from the PDU to a file named "diag-data.tgz."

SCP example:

```
scp  admin@192.168.87.50:/diag-data.tgz
```

Windows PSCP command:

PSCP in Windows works in a similar way to the SCP.

```
pscp <user name>@<device ip>:/diag-data.tgz
```

Chapter 7: Enabling Service Advertising

The PDU advertises all enabled services that are reachable using the IP network. This feature uses DNS-SD (Domain Name System-Service Discovery) and MDNS (Multicast DNS). The advertised services are discovered by clients that have implemented DNS-SD and MDNS.

The advertised services include the following:

- HTTP
- HTTPS
- Telnet
- SSH
- Modbus
- json-rpc
- SNMP

By default, this feature is enabled.

Enabling this feature also enables Link-Local Multicast Name Resolution (LLMNR) and/or MDNS, which are required for resolving APIPA host names. See ***APIPA and Link-Local Addressing*** (on page 152).

The service advertisement feature supports both IPv4 and IPv6 protocols.

If you have set a preferred host name for IPv4 and/or IPv6, that host name can be used as the zero configuration .local host name, that is, *<preferred_host_name>.local*, where <preferred_host_name> is the preferred host name you have specified for the PDU. The IPv4 host name is the first priority. If an IPv4 host name is not available, then use the IPv6 host name.

Note: For information on configuring IPv4 and/or IPv6 network settings, refer to ***Wired Network Settings*** *in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.*

To enable or disable service advertising:

1. Choose Device Settings > Network Services > Service Advertising.
2. To enable the service advertising, select either or both checkboxes.
   - To advertise via MDNS, select the Multicast DNS checkbox.
   - To advertise via LLMNR, select the Link-Local Multicast Name Resolution checkbox.
3. Click Save.

APIPA and Link-Local Addressing

The PDU supports Automatic Private Internet Protocol Addressing (APIPA).

With APIPA, your PDU automatically configures a link-local IP address and a link-local host name when it cannot obtain a valid IP address from any DHCP server in the TCP/IP network.

Only IT devices connected to *the same subnet* can access the PDU using the link-local address/host name. Those in a different subnet cannot access it.

Exception: The PDU in the Port Forwarding mode does not support APIPA.

Once the PDU can get a DHCP-assigned IP address, it stops using APIPA and the link-local address is replaced by the DHCP-assigned address.

Scenarios where APIPA applies:

- DHCP is enabled on the PDU, but no IP address is assigned to the PDU.

  This may be caused by the absence or malfunction of DHCP servers in the network.

  Note: Configuration by connecting the PDU to a computer using a network cable is an application of this scenario. See ***Connecting the PDU to a Computer*** *(on page 7).*

- The PDU previously obtained an IP address from the DHCP server, but the lease of this IP address has expired, and the lease cannot be renewed, or no new IP address is available.

Link-local addressing:

- IPv4 address:

  Factory default is to enable IPv4 only. The link-local IPv4 address is *169.254.x.x/16*, which ranges between 169.254.1.0 and 169.254.254.255.

- IPv6 address:

  A link-local IPv6 address is available only after IPv6 is enabled on the PDU. Refer to ***Configuring Network Settings*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.

- Host name - **pdu.local**:

  You can type *https://pdu.local* to access the PDU instead of typing the link-local IP address.

  For retrieval of the link-local address, refer to ***Device Info*** in the Premium+ PDU With RackLink User Manual at www.middleatlantic.com/resources/power-downloads.aspx.

Chapter 8: Troubleshooting

Windows NTP Server Synchronization Solution

The NTP client on the PDU follows the NTP RFC so the PDU rejects any NTP servers whose root dispersion is more than one second. An NTP server with a dispersion of more than one second is considered an inaccurate NTP server by the PDU.

Note: For information on NTP RFC, refer to section 5 at http://tools.ietf.org/html/rfc4330.

Windows NTP servers may have a root dispersion of more than one second, and therefore cannot synchronize with the PDU. When the NTP synchronization issue occurs, change the dispersion settings to resolve it.

To change the Windows NTP's root dispersion settings:

1. Access the registry settings associated with the root dispersion on the Windows NTP server.

   HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W32Time\Config

2. *AnnounceFlags* must be set to 0x05 or 0x06.
   - 0x05 = 0x01 (Always time server) and 0x04 (Always reliable time server)
   - 0x06 = 0x02 (Automatic time server) and 0x04 (Always reliable time server)

   Note: Do NOT use 0x08 (Automatic reliable time server) because its dispersion starts at a high value and then gradually decreases to one second or lower.

3. *LocalClockDispersion* must be set to 0.

Chapter 9: Compliance with IEC 62020

IEC 62020 is an international standard for Residual Current Monitors. All PDUs with RCM are IEC 62020 compliant.

IEC 62020 uses the term *rated residual operating current* (IΔn) to specify residual current, equal to or above which causes an alarm. IEC 62020 recommends preferred values 6mA, 10mA, 30mA, 100mA, 300mA and 500mA. In the PDU with RCM, IΔn is specified using the Critical Rated Residual Operating Current threshold.

Note: The PDU triggers events when residual current values are above (but not equal to) thresholds. For example, you would set the critical threshold to 29mA to specify the IEC 62020 IΔn of 30mA.

IEC 62020 uses the term *residual non-operating current* (IΔno) to specify residual current, below which does not cause an alarm. IEC 62020 specifies IΔno be no higher than 0.5 IΔn. In the PDU with RCM, IΔno is set using the RCM Deassertion Hysteresis and this value must be no higher than 0.5 the RCM critical threshold.

The PDU with RCM allows you to establish an optional WARNING state, which is not part of the IEC 62020 specification. The PDU RCM remains IEC 62020 compliant when the RCM deassertion hysteresis is configured properly.

IEC 62020 specification	PDU with RCM characteristics
Method of operation	Dependent on line voltage. RCM only functions if line voltage is present.
Type of installation	PDU with flexible line cords and plugs are for mobile installation and corded connection.
Current paths	1-phase PDU are two current paths RCM.
Ability to adjust residual operating current	Adjustable. ▪ Type A: 6mA-500mA. ▪ Type B: 30mA-500mA.
Adjustable time delay	Non-adjustable time delay.
Protection against external influence	Enclosed-type RCM.
Method of mounting	Panel board type RCM.
Method of connection	Not associated with mechanical mounting.
Connection of load conductors	Monitored line is directly connected.
Fault indicating means	Visual, with other output signals.
Ability to directly discriminate	Directionally non-discriminating.

IEC 62020 specification	PDU with RCM characteristics
Rated residual operating current	0.5A (highest value).
Residual currents with direct current components	Model dependent. Models ending in -M5 are Type A, -M11 are Type B.

WARRANTY

For warranty information, refer to http://www.middleatlantic.com/company/about-us.aspx#warranty.

Corporate Headquarters
Voice: 973-839-1011 – Fax: 973-839-1976 – International Voice: +1 973-839-8821 –
Fax: +1 973-839-4982 – www.middleatlantic.com – info@middleatlantic.com

Middle Atlantic Canada
Voice: 613-836-2501 – Fax: 613-836-2690 – ca.middleatlantic.com – customerservicecanada@middleatlantic.ca

Middle Atlantic EMEA Technical Support
Voice: +31 (0) 495 726002 - av.emea.middleatlantic.support@legrand.com

Factory Distribution
United States: New Jersey, California, Illinois - Canada: Ontario - The Netherlands: Weert

At Middle Atlantic Products we are always listening. Your comments are welcome.

Middle Atlantic Products is an ISO 9001 and ISO 14001 Registered Company.