

Introduction

This chapter contains the following sections:

- Product Notes, on page 2
- Overview, on page 3
- User Privilege Levels, on page 4
- CLI Command Modes, on page 5
- Interfaces for Debug Access, on page 7
- Accessing the CLI, on page 8
- CLI Command Conventions, on page 10
- Editing Features, on page 11
- Interface Naming Conventions, on page 13
- IPv6z Address Conventions, on page 15
- Loopback Interface, on page 16
- Managing Ports Via CLI, on page 18
- Remote IP Address and OOB Port, on page 19
- PHY Diagnostics, on page 20
- CLI Output Modifiers, on page 21

Product Notes

This CLI guide provides CLI commands and guidelines for the CBS 350 Stacking and Non-Stacking switches. These product line supports 2 "sub-types" - the devices in the first sub-type support 10GigabitEthernet on all ports, the devices of the 2nd subtype support 10GigabitEthernet only on the uplink ports. Besides a few CLI commands, which will be mentioned below, the CLI commands included in this document can be applied to both "subtypes". Following are the notes and differences in CLI command support in regards to these product lines:

- Ports types—
  - The "all 10G" port subtype support ports with TengigabitEthernet (XG) speed.
  - The "10G uplink" port subtype can support different network ports type of: Gigabitethernet (GE), 2.5 Gigabit Ethernet (TW) and 5 Gigabit Ethernet (FI); In addition these devices support 4 XG uplink ports.

  CLI examples in this document use the GE port type in examples, but the same commands can be applied to TW, FI or XG port types, unless there is a difference in feature implementation between port types.
- Speed and negotiation settings—Each port type supports the negotiation and speed setting relevant to the port type. For example a GE interface does not support speed or negotiation of 10G interface.
- OOB interface—The "all 10G" port subtype supports an OOB interface, while the "10G uplink port subtype does not. Therefore, OOB as a configurable interface is relevant only to the "all 10G" port subtype. For devices that support OOB – DHCP client and default IP address (192.168.1.254) are applied to the OOB port and not to the default VLAN.
- Power Over Ethernet—PoE is supported on some of the "10G uplink" port subtype devices and not on the "all 10G" port subtype devices. Therefore PoE commands are relevant only to the "10G uplink" port subtype.
- Stacking—On both subtypes only 10G interfaces can be defined as stacking interfaces. Short reach and energy detect—Short Reach is always enabled on the "all 10G" port subtype devices and TW and FI ports (all SKUs); Energy Detect is always enabled on XG ports of the "all 10G" port subtype devices; For all other interface types- both of these features can be enabled or disabled (default is disabled).
- MAC address aging time—The maximum value for "all 10G" port subtype devices is 630 seconds, while the maximum value for "10G uplink" port subtype devices and Hybrid mode stack is 400 seconds. The default value for both subtypes is the same – 300 seconds.
- IPv6 tunnels—IPv6 Manual, 6to4 and ISATAP routing tunnels are supported on the "all 10G" port subtype devices, and are not supported on "10G uplink" port subtype devices.
- System Router Resources—The default values stated in command are for the "10G uplink" port subtype devices. The "all 10G" port subtype devices supports the following default values:
  - policy-ip-entries—16
  - policy-ipv6-entries—16
  - VLAN mapping entries —32

Overview

The CLI is divided into various command modes. Each mode includes a group of commands. These modes are described in CLI Command Modes, on page 5.

Users are assigned privilege levels. Each user privilege level can access specific CLI modes. User levels are described in the section below.

User Privilege Levels

Users can be created with one of the following user levels:

- Level 1—Users with this level can only run User EXEC mode commands. Users at this level cannot access the web GUI or commands in the Privileged EXEC mode.
- Level 7—Users with this level can run commands in the User EXEC mode and a subset of commands in the Privileged EXEC mode. Users at this level cannot access the web GUI.
- Level 15—Users with this level can run all commands. Only users at this level can access the web GUI.

A system administrator (user with level 15) can create passwords that allow a lower level user to temporarily become a higher level user. For example, the user may go from level 1 to level 7, level 1 to 15, or level 7 to level 15.

The passwords for each level are set (by an administrator) using the following command:

enable password [level privilege-level]{password|encrypted encrypted-password}

Using these passwords, you can raise your user level by entering the command: enable and the password for level 7 or 15. You can go from level 1 to level 7 or directly to level 15. The higher level holds only for the current session.

The disable command returns the user to a lower level.

To create a user and assign it a user level, use the username command. Only users with command level 15, can create users at this level.

Example—Create passwords for level 7 and 15 (by the administrator):

```
switchxxxxxx#configure
switchxxxxxx<conf># enable password level 7 level7@aBc
switchxxxxxx<conf># enable password level 15 level15@aBc
switchxxxxxx<conf>#
```

Create a user with user level 1:

```
switchxxxxxx#configure
switchxxxxxx<conf> username john password John1234 privilege 1
switchxxxxxx<conf>
```

Example 2— Switch between Level 1 to Level 15. The user must know the password:

```
switchxxxxxx#
switchxxxxxx# enable
Enter Password: ****** (this is the password for level 15
- Level15@abc)
switchxxxxxx#
```

Note If authentication of passwords is performed on RADIUS or TACACS+ servers, the passwords assigned to user level 7 and user level 15 must be configured on the external server and associated with the $enable7$ and $enable15$ user names, respectively

CLI Command Modes

The CLI is divided into four command modes. The command modes are (in the order in which they are accessed):

- User EXEC mode
- Privileged EXEC mode
- Global Configuration mode

Each command mode has its own unique console prompt and set of CLI commands. Entering a question mark at the console prompt displays a list of available commands for the current mode and for the level of the user. Specific commands are used to switch from one mode to another.

Users are assigned privilege levels that determine the modes and commands available to them.

User EXEC Mode

Users with level 1 initially log into User EXEC mode. User EXEC mode is used for tasks that do not change the configuration, such as performing basic tests and listing system information.

The user-level prompt consists of the switch host name followed by a #. The default host name is switchxxxxxx where xxxxxx is the last six digits of the device's MAC address, as shown below

switchxxxxxx#

The default host name can be changed via the hostname command in Global Configuration mode.

Privileged EXEC Mode

A user with level 7 or 15 automatically logs into Privileged EXEC mode.

Users with level 1 can enter Privileged Exec mode by entering the enable command, and when prompted, the password for level 15.

To return from the Privileged EXEC mode to the User EXEC mode, use the disable command

Global Configuration Mode

The Global Configuration mode is used to run commands that configure features at the system level, as opposed to the interface level.

Only users with command level of 7 or 15 can access this mode.

To access Global Configuration mode from Privileged EXEC mode, enter the configure command at the Privileged EXEC mode prompt and press Enter. The Global Configuration mode prompt, consisting of the device host name followed by (config)#, is displayed:

switchxxxxxx(config)#

Use any of the following commands to return from Global Configuration mode to the Privileged EXEC mode:

- exit
- end
- Ctrl+Z

The following example shows how to access Global Configuration mode and return to Privileged EXEC mode:

```
switchxxxxxx#
switchxxxxxx# configure
switchxxxxxx(config)# exit
switchxxxxxx#
```

Interface or Line Configuration Modes

Various submodes may be entered from Global Configuration mode. These submodes enable performing commands on a group of interfaces or lines.

For instance to perform several operations on a specific port or range of ports, you can enter the Interface Configuration mode for that interface.

The following example enters Interface Configuration mode for vlan1 and then sets their speed:

The exit command returns to Global Configuration mode.

```
switchxxxxxx#
switchxxxxxx# configure
switchxxxxxx(config)# interface range vlan1
switchxxxxxx(config-if)#speed 10
switchxxxxxx(config-if)#exit
switchxxxxxx(config)#
```

The following is a sample of some of the available submodes:

- Interface—Contains commands that configure a specific interface (port, VLAN, port channel, or tunnel) or range of interfaces. The Global Configuration mode command interface is used to enter the Interface Configuration mode. The interface Global Configuration command is used to enter this mode.
- Line Interface—Contains commands used to configure the management connections for the console, Telnet and SSH. These include commands such as line timeout settings, etc. The line Global Configuration command is used to enter the Line Configuration command mode.
- VLAN Database—Contains commands used to configure a VLAN as a whole. The vlan database Global Configuration mode command is used to enter the VLAN Database Interface Configuration mode.
- Management Access List—Contains commands used to define management access-lists. The management access-list Global Configuration mode command is used to enter the Management Access List Configuration mode.
- MAC Access-List, IPv6 Access List, IP Access List—Configures conditions required to allow traffic based on MAC addresses, IPv6 address and IPv4 address, respectively. The mac access-list, ipv6 access-list and ip access-list Global Configuration mode commands are used to enter the these configuration mode.

To return from any Interface Configuration mode to the Global Configuration mode, use the exit command.

Interfaces for Debug Access

In addition to the standard CLI interface modes detailed above, the device supports additional interfaces for device debug access. These interlaces are intended to be used by a Cisco Support Team personnel, in cases where it is required to debug device's behavior. These interfaces are password protected. The passwords are held by the Cisco support team.

The device supports the following debug interfaces:

- U-BOOT access during boot sequence (access is possible only via serial console terminal)
- Linux Kernel access during boot sequence (access is possible only via serial console terminal)
- Run time debug modes - allows Cisco Support Team personnel to view device settings, and to apply protocol and layer 1 debug commands and settings (access is possible via serial, telnet or SSH console terminal)

Accessing the CLI

The CLI can be accessed from a terminal or computer by performing one of the following tasks:

- Running a terminal application, such as HyperTerminal, on a computer's com port that is directly connected to the switch's console port, or
- Running a Telnet session from a command prompt on a computer with a network connection to the switch.
- Using SSH from an application that supports SSH client running on a computer with a network connection to the switch.



Note Telnet and SSH are disabled by default on the switch.

If access is via a Telnet or SSH connection, ensure that the following conditions are met before using CLI commands:

- The switch has a defined IP address
- Corresponding management access is enabled.
- There is an IP path such that the computer and the switch can reach each other

Using HyperTerminal over the Console Interface

Some of the devices in this product line support a single RJ45 console management interface, while other devices in this product line supports a dual console management interface - mini USB and an RJ45 port. The RJ45 interface provides a direct connection to a computer's serial port using a standard DB-9 null-modem or crossover cable. If Both min USB and RJ45 are connected, on the devices that support dual management interfaces, the mini USB interface has precedence



Note Mini USB interface will become active a few seconds after device is turned on/rebooted

After the computer and switch are connected, run a terminal application to access the CLI. The terminal emulator must be configured to databits=8 and parity=none.

Click **Enter** twice, so that the device sets the serial port speed to match the PC's serial port speed.

When the CLI appears, enter cisco at the User Name prompt and then enter cisco for the Password prompt.



Note If this is the first time that you have logged on with the default username and password, the device will display a prompt to change username and Password. The new password needs to comply to password complexity rules.

The switchxxxxxx# prompt is displayed. You can now enter CLI commands to manage the switch. For detailed information on CLI commands, refer to the appropriate chapter(s) of this reference guide.

Using Telnet over an Ethernet Interface

Telnet provides a method of connecting to the CLI over an IP network.

To establish a telnet session from the command prompt, perform the following steps

Procedure

Step 1 Click **Start,** then select **All Programs > Accessories > Command Prompt** to open a command prompt.

Step 2 At the prompt, enter **telnet 1<IP address of switch**>, then press **Enter**.

Step 3 CLI will be displayed..

Step 4 When the CLI appears, enter the defined username at the User Name prompt and then enter the defined password at the Password prompt.

The switchxxxxxx# prompt is displayed. You can now enter CLI commands to manage the switch.

For detailed information on CLI commands, refer to the appropriate chapter(s) of this reference guide.

CLI Command Conventions

When entering commands there are certain command entry standards that apply to all commands. The following table describes the command conventions.

Convention	Description
[ ]	In a command line, square brackets indicate an optional entry.
{ }	In a command line, curly brackets indicate a selection of compulsory parameters separated the | character. One option must be selected. For example, **flowcontrol {auto|on|off}** means that for the flowcontrol command, either auto, on, or off must be selected
"" (inverted commas)	When the input string contains space and/or reserved words (i.e. VLAN), put the string in inverted commas.
parameter	Italic text indicates a parameter.
press key	Names of keys to be pressed are shown in bold.
Ctrl+F4	Keys separated by the + character are to be pressed simultaneously on the keyboard
Screen Display	Fixed-width font indicates CLI prompts, CLI commands entered by the user, and system messages displayed on the console.
all	When a parameter is required to define a range of ports or parameters and all is an option, the default for the command is all when no parameters are defined. For example, the command interface range port-channel has the option of either entering a range of channels, or selecting all. When the command is entered without a parameter, it automatically defaults to all.
text	When free text can be entered as a parameter for a command (for example in command: snmp-server contact) if the text consists of multiple words separated by blanks, the entire string must appear in double quotes. For example: **snmp-server contact** "QA on floor 8"

Editing Features

Entering Commands

A CLI command is a series of keywords and arguments. Keywords identify a command, and arguments specify configuration parameters. For example, in the command show interfaces status Gigabitethernet 1, show, interfaces and status are keywords, Gigabitethernet is an argument that specifies the interface type, and1 specifies the port.

To enter commands that require parameters, enter the required parameters after the command keyword. For example, to set a password for the administrator, enter:

switchxxxxxx(config)# username admin password Alansmith1

When working with the CLI, the command options are not displayed. The standard command to request help is ?

There are two instances where help information can be displayed:

- Keyword lookup—The character ? is entered in place of a command. A list of all valid commands and corresponding help messages are is displayed
- Partial keyword lookup—If a command is incomplete and or the character ? is entered in place of a parameter, the matched keyword or parameters for this command are displayed.

Terminal Command Buffer

Every time a command is entered in the CLI, it is recorded on an internally managed Command History buffer. Commands stored in the buffer are maintained on a First In First Out (FIFO) basis. These commands can be recalled, reviewed, modified, and reissued. This buffer is not preserved across device resets.

Keyword	Description
Up-Arrow key	Recalls commands in the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Down-Arrow key	Returns to more recent commands in the history buffer after recalling commands with the up-arrow key. Repeating the key sequence will recall successively more recent commands.

By default, the history buffer system is enabled, but it can be disabled at any time. For more information on enabling or disabling the history buffer, refer to the history command.

There is a standard default number of commands that are stored in the buffer. The standard number of 10 commands can be increased to 216. By configuring 0, the effect is the same as disabling the history buffer system. For more information on configuring the command history buffer, refer to the **history size** command.

To display the history buffer, refer to the **show history** command.

Negating the Effect of Commands

For many configuration commands, the prefix keyword no can be entered to cancel the effect of a command or reset the configuration to the default value. This Reference Guide provides a description of the negation effect for each CLI command.

Command Completion

If the command entered is incomplete, invalid or has missing or invalid parameters, then the appropriate error message is displayed. This assists in entering the correct command. By pressing Tab after an incomplete command is entered, the system will attempt to identify and complete the command. If the characters already entered are not enough for the system to identify a single matching command, press ? to display the available commands matching the characters already entered.

Keyboard Shortcuts

The CLI has a range of keyboard shortcuts to assist in editing the CLI commands. The following table describes the CLI shortcuts.

Keyboard Key	Description
Up-arrow	Recalls commands from the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Down-arrow	Returns the most recent commands from the history buffer after recalling commands with the up arrow key. Repeating the key sequence will recall successively more recent commands.
Ctrl+A	Moves the cursor to the beginning of the command line.
Ctrl+E	Moves the cursor to the end of the command line.
Ctrl+Z / End	Returns back to the Privileged EXEC mode from any configuration mode.
Backspace	Deletes one character left to the cursor position.

Copying and Pasting Text

Up to 1000 lines of text (or commands) can be copied and pasted into the device.



Note It is the user's responsibility to ensure that the text copied into the device consists of legal commands only.

When copying and pasting commands from a configuration file, make sure that the following conditions exist:

- A device Configuration mode has been accessed.

The commands contain no encrypted data, like encrypted passwords or keys. Encrypted data cannot be copied and pasted into the device except for encrypted passwords where the keyword encrypted is used before the encrypted data (for instance in the enable password command).

Interface Naming Conventions

Interfaces on the device can be one of the following types:

- Gigabit Ethernet (10/100/1000 kbits) ports—These can be written as either GigabitEthernet or gi or GE.
- 2.5 Gigabit Ethernet (10/100/1000/25000 kbits) ports—These can be written as either TwoPointFiveGigabitEthernet or tw.
- 5 Gigabit Ethernet (10/100/1000/25000/50000 kbits) ports—These can be written as either FiveGigabitEthernet or fi.
- LAG (Port Channel)—Written as either Port-Channel or po.
- VLAN—Written as VLAN
- Tunnel—Written as tunnel or tu
- OOB—Written as OutOfBand or oob

Within the CLI, interfaces are denoted by concatenating the following elements:

- Type of Interface—As described above
- Unit Number—Unit in stack.
- Slot Number—The slot number is always 0.
- The syntax for interface names in stacking mode is:

  {<port-type>[ ][<unit-number>/]<slot-number>/<port-number>} | {port-channel | po |

  }[ ]<port-channel-number> |

  {tunnel | tu}[ ]<tunnel-number> | vlan[ ]<vlan-id>
- Interface Number—Port, LAG, tunnel or VLAN numbers

Samples of these various options are shown in the example below:

```
switchxxxxxx(config)#interface GigabitEthernet 1
switchxxxxxx(config)#interface GE 1
switchxxxxxx(config)#interface TwoPointFiveGigabitEthernet
switchxxxxxx(config)#interface po1
switchxxxxxx(config)# interface vlan 1
```

Interface Range

Interfaces may be described on an individual basis or within a range. The interface range command has the following syntax:

<interface-range> ::=

{<port-type>[

][<unit-number>/]<slot-number>/<first-port-number>[-

<last-port-number]} |

port-channel[ ]<first-port-channel-number>[-

<last-port-channel-number>] |

tunnel[ ]<first-tunnel-number>[- <last-tunnel-number>] |

vlan[ ]<first-vlan-id>[- <last-vlan-id>]

A sample of this command is shown in the example below:

```
switchxxxxxx#configure
switchxxxxxx(config-if)#interface range gi1-5g
```

List of Multiple Interface Types

A combination of interface types can be specified in the interface range command in the following format:

<range-list> ::= <interface-range> | <range-list>, <interface-range>

Up to five ranges can be included.

Note Range lists can contain either ports and port-channels or VLANs. Combinations of port/port-channels and VLANs are not allowed.

The space after the comma is optional.

When a range list is defined, a space after the first entry and before the comma (,) must be entered.

A sample of this command is shown in the example below:

```
switchxxxxxx#configure
switchxxxxxx(config)#interface range gi1-5, vlan 1-2
```

IPv6z Address Conventions

The following describes how to write an IPv6z address, which is a link-local IPv6 address.

The format is: <ipv6-link-local-address>%<egress-interface>

where:

egress-interface (also known as zone) = vlan<vlan-id> | po<number> | tunnel<number> | port<number> | 0

If the egress interface is not specified, the default interface is selected. Specifying egress interface = 0 is equal to not defining an egress interface.

The following combinations are possible:

- ipv6_address%egress-interface—Refers to the IPv6 address on the interface specified.
- ipv6_address%0—Refers to the IPv6 address on the single interface on which an IPv6 address is defined.
- ipv6_address—Refers to the IPv6 address on the single interface on which an IPv6 address is defined.

Loopback Interface

When an IP application on a router wants to communicate with a remote IP application, it must select the local IP address to be used as its IP address. It can use any IP address defined on the router, but if this link goes down, the communication is aborted, even though there might well be another IP route between these IP applications.

The loopback interface is a virtual interface whose operational state is always up. If the IP address that is configured on this virtual interface is used as the local address when communicating with remote IP applications, the communication will not be aborted even if the actual route to the remote application was changed.

The name of the loopback interface is loopback1.

A loopback interface does not support bridging; it cannot be a member of any VLAN, and no layer 2 protocol can be enabled on it.

Layer 3 Specification

IP Interface

IPv4 and IPv6 addresses can be assigned to a loopback interface.

The IPv6 link-local interface identifier is 1.

Routing Protocols

A routing protocol running on the switch supports the advertising of the IP prefixes defined on the loopback interfaces via the routing protocol redistribution mechanism.

Configuration Examples

Static Routing

The following example shows you how to configure IP on a switch with static routing:

```
Switch# configure terminal
Switch(config)# interface vlan 1
Switch(config-if)# ip address 10.10.10.2 /24
Switch(config-if)# ipv6 address 2001:DB8:2222:7270::2312/64
Switch(config-if)# exit
Switch(config)# interface vlan 2
Switch(config-if)# ip address 10.11.11.2 /24
Switch(config-if)# ipv6 address 2001:DB8:3333:7271::2312/64
Switch(config-if)# exit
Switch(config)# interface loopback 1
Switch(config-if)# ip address 172.25.13.2 /32
Switch(config-if)# ipv6 address 2001:DB8:2222:7272::72/128
Switch(config-if)# exit
Switch(config)# ip route 0.0.0.0/0 10.10.11.1
Switch(config)# ip route 10.11.0.0 /16 10.11.11.1
Switch(config)# ipv6 route 0::/0 2001:DB8:2222:7270::1
Switch(config)# ipv6 route 2001:DB8:3333::/48
2001:DB8:3333:7271::1
```

The neighbor router 10.10.11.1 should be configured with the following static route: ip route 172.25.13.2 /32 10.10.10.2.

The neighbor router 10.11.11.1 should be configured with the following static route: ip route 172.25.13.2 /32 10.11.11.2.

The neighbor router 2001:DB8:2222:7270::1 connected to VLAN 1 should be configured with the following static route:

ipv6 route 2001:DB8:2222:7272::72/128 2001:DB8:2222:7270::2312

The neighbor router 2001:DB8:3333:7271::1 connected to VLAN 1 should be configured with the static route defined immediately below.

IPv6 Route 2001:DB8:2222:7272::72/128 2001:DB8:3333:7271::2312

Managing Ports Via CLI

To access a port interface on units that support stacking, type “interfaceGigabitEthernetX/0/Z (for 1gig interfaces), or “interface TenGigabitEthernetX/0/Y for 10gig ports with X (1-4) being the stack ID, Y for the uplink ports number (1-4), and Z for the downlink port number; Z is between 1- 48 even for the units that have less than 48 ports.

Fiber cable and Transceivers

Cisco brand provides a panoply a SFP modules, while our switches support other 3rd party, it is important to pay attention on the type of fiber cable to use in conjunction with the specific SFP module.

Fiber cables can be classified in two types: Single Mode and Multimode. The main difference is the distance they are capable of covering and their diameter. Single Mode fibers cover greater distance compared to Multimode and has lower diameter (around 9 micrometer) while the Multimode fibers diameter is 50-62.5 micrometer.

DAC cables for Direct Attach Copper Cable, on the other hand, can be used for short distances. They are mainly based on the Multimode-Standard type of transceivers due to the fact that the max distance they can cover is 15m. The AOC (Active Optical Cable), however, is a different story.

When troubleshooting fiber connectivity related issues, it is important to make a distinction between SMF (Single Mode Fiber cables), and MMF (Multimode Fiber cables) and their corresponding SFP transceivers a given fiber can support.

Cisco has matrix that we can refer to when trying to make a determination of the correct pairing. The following link provide some insight on Cisco 10gig SFP

Example:

The Cisco SFP-10G-SR only works with MMF type of cable of 62.5 micrometer diameter, while the Cisco SFP-10G-LR works only with SMF type of cable. The MMF range from OM1-OM5. OM is for Optical Multimode. OM1 type cables have a diameter of 62.5 micrometer, while all other types (OM2-OM5) have a diameter of 50 micrometer.

So, it is important to know what is being done to avoid mixing them up.

Remote IP Address and OOB Port

The switch supports an IP stack on the OutOfBand (OOB) port. This IP stack is separate from the IP stack running on the ASIC ports, and it requires specific route table configuration

If the switch supports more than one IP interface, when you specify a remote IP address or a DNS name, you must also specify the IP stack that is being referred to.

PHY Diagnostics

The following exceptions exist:

- Copper Ports—PHY diagnostics are only supported on copper ports.
- 10G ports—TDR test is supported when the operational port speed is 10G. Cable length resolution is 20 meters.

CLI Output Modifiers

To all **show** and **more** commands (except **show technical support**) an output modifier may be added as follows:

```
<show/more command> | <output-modifier> <regular-expression-pattern>
```

The output modifiers are:

- **begin:** Start output from the first line that has a sequence of characters matching thegiven regular expression pattern
- **include:** Includes only lines that have a sequence of characters matching the given regular expression pattern.
- **exclude:** Excludes all lines that have a sequence of characters matching the given regular expression pattern.
- **count:** Counts all lines that have a sequence of characters matching the given regular expression pattern and displays the result (no other output is displayed).



Note Only 1 output modifier can be used in each command. The remainder of the text typed in is part of the regular expression pattern.

A regular expression is a pattern (a phrase, number, or more complex pattern). The CLI String Search feature matches regular expressions to the show or more command output. Regular expressions are case-sensitive and allow for complex matching requirements.

A regular expression can be a single-character pattern or a multiple-character pattern. That is, a regular expression can be a single character that matches the same single character in the command output or multiple characters that match the same multiple characters in the command output. The pattern in the command output is referred to as a string. This section describes creating both single-character patterns and multiple-character patterns. It also discusses creating more complex regular expressions, using multipliers, alternation, anchoring, and parentheses.

Single-Character Patterns

The simplest regular expression is a single character that matches the same single character in the command output. You can use any letter (A-Z, a-z) or digit (0-9) as a single-character pattern. You can also use other keyboard characters (such as ! or ~) as single-character patterns, but certain keyboard characters have special meaning when used in regular expressions. The following table lists the keyboard characters that have special meanings

Character	Meaning
.	Matches any single character, including white space.
*	Matches 0 or more sequences of the pattern.
+	Matches 1 or more sequences of the pattern.
?	Matches 0 or 1 occurrences of the pattern.

Character	Meaning
^	Matches the beginning of the string.
$	Matches the end of the string.

To use these special characters as single-character patterns, remove the special meaning by preceding each character with a backslash (\).

The following examples are single-character patterns matching a dollar sign, an underscore, and a plus sign, respectively.

\$ _ \+

You can specify a range of single-character patterns to match against command output. For example, you can create a regular expression that matches a string containing one of the following letters: a, e, i, o, or u. Only one of these characters must exist in the string for pattern matching to succeed. To specify a range of single-character patterns, enclose the single-character patterns in square brackets ([ ]). For example, [aeiou] matches any one of the five vowels of the lowercase alphabet, while [abcdABCD] matches any one of the first four letters of the lower- or uppercase alphabet.

You can simplify ranges by entering only the endpoints of the range separated by a dash (-).

Simplify the previous range as follows:

[a-dA-D]

To add a dash as a single-character pattern in your range, include another dash and precede it with a backslash:

[a-dA-D\-]

You can also include a right square bracket (]) as a single-character pattern in your range, as shown here:

[a-dA-D\-\]]

The previous example matches any one of the first four letters of the lower- or uppercase alphabet, a dash, or a right square bracket. You can reverse the matching of the range by including a caret (^) at the start of the range. The following example matches any letter except the ones listed:

[^a-dqsv]

The following example matches anything except a right square bracket (]) or the letter d:

[^\]d]

Multiple-Character Patterns

When creating regular expressions, you can also specify a pattern containing multiple characters. You create multiple-character regular expressions by joining letters, digits, or keyboard characters that do not have special meaning. For example, a4% is a multiple-character regular expression.

With multiple-character patterns, order is important. The regular expression a4% matches the character a followed by a 4 followed by a % sign. If the string does not have a4%, in that order, pattern matching fails. The multiple-character regular expression a. uses the special meaning of the period character to match the letter a followed by any single character. With this example, the strings ab, a!, or a2 are all valid matches for the regular expression.

You can remove the special meaning of the period character by inserting a backslash before it. For example, when the expression a\. is used in the command syntax, only the string a. will be matched.

You can create a multiple-character regular expression containing all letters, all digits, all keyboard characters, or a combination of letters, digits, and other keyboard characters. For example, telebit 3107 v32bis is a valid regular expression.

Multipliers

You can create more complex regular expressions that instruct the system to match multiple occurrences of a specified regular expression. To do so, use some special characters with your single-character and multiple-character patterns. Table 1 lists the special characters that specify multiples of a regular expression.

Table 1: Table 1: Special Characters Used as Multipliers

Character	Description
*	Matches 0 or more single-character or multiple-character patterns.
+	Matches 1 or more single-character or multiple-character patterns.
?	Matches 0 or 1 occurrences of a single-character or multiple-character pattern.

The following example matches any number of occurrences of the letter a, including none:

a*

The following pattern requires that at least one letter a be in the string to be matched:

a+

The following pattern matches the string bb or bab:

ba?b

The following string matches any number of asterisks (*):

**

To use multipliers with multiple-character patterns, enclose the pattern in parentheses. In the following example, the pattern matches any number of the multiple-character string ab:

(ab)*

The following pattern matches one or more instances of alphanumeric pairs, but not none (that is, an empty string is not a match):

([A-Za-z][0-9])+

The order for matches using multipliers (*, +, or ?) is to put the longest construct first. Nested constructs are matched from outside to inside. Concatenated constructs are matched beginning at the left side of the construct. Thus, the regular expression above matches A9b3, but not 9Ab3 because the letters are specified before the numbers.

Alternation

Alternation allows you to specify alternative patterns to match against a string. You separate the alternative patterns with a vertical bar (|). Only one of the alternatives can match the string. For example, the regular expression codex|telebit either matches the string codex or the string telebit, but not both codex and telebit.

Anchoring

You can instruct the system to match a regular expression pattern against the beginning or the end of the string. You anchor these regular expressions to a portion of the string using the special characters shown in Table 2.

Table 2: Table 2: Special Characters Used for Anchoring

Character	Description
^	Matches the beginning of the string.
$	Matches the end of the string.

For example, the regular expression ^con matches any string that starts with con, and $sole matches any string that ends with sole.

In addition to indicating the beginning of a string, the ^ symbol can be used to indicate the logical function not when used in a bracketed range. For example, the expression [^abcd] indicates a range that matches any single letter, as long as it is not the letters a, b, c, or d.