

04 The IOS Operating System - Lab Exercise

This lab explores basic navigation of the Cisco IOS operating system CLI (Command Line Interface). Only a single device is required.

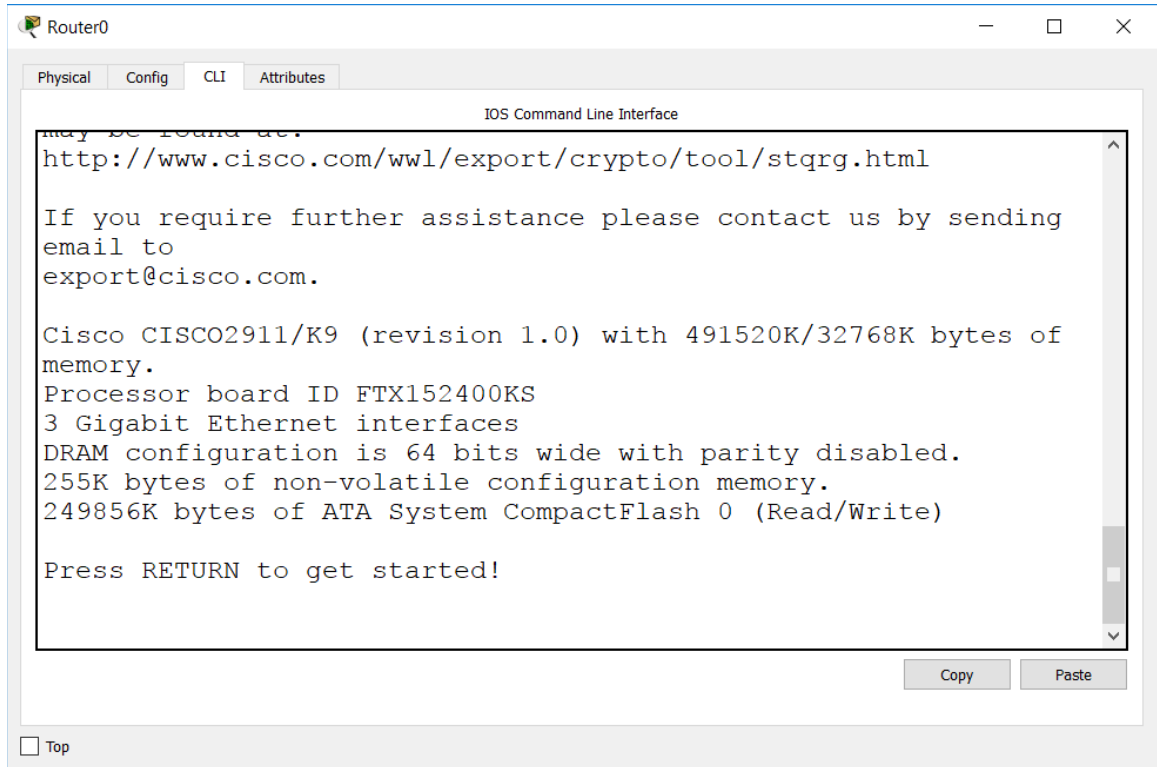
This lab is a guided walkthrough of the IOS command line interface. Exercises for later sections will be split into two parts - first the tasks for you to complete on your own (without step by step instructions), and then an answer key showing you the solution.

Load the Startup Configuration

Open the '04 The IOS Operating System.pkt' file in Packet Tracer to load the lab.

Connect To Your Device

Click on Router0 and then the CLI tab to access the console.



Press Return to get started, then enter Privileged Exec mode.

```
Router>enable
Router#
```

Reboot the device.

```
Router#reload
Proceed with reload? [confirm]
```

Observe the device going through the bootup process in the command line output. This is possible because we are using a console connection (we could not see this if we connected to an IP address on the device.)

If prompted to enter the initial configuration dialog after the device has booted up, enter 'no'.

```
Would you like to enter the initial configuration dialog?
[yes/no]: no
```

Explore User Exec Mode and CLI command help

Notice that you are in User Exec mode as indicated by the 'Router>' prompt. ('Router' will be replaced with the device hostname after you configure one.)

```
Router>
```

Enter a question mark to explore the commands that are available in User Exec mode.

```
Router>?  
Exec commands:  
<1-99> Session number to resume  
connect Open a terminal connection  
disable Turn off privileged commands  
disconnect Disconnect an existing network connection  
enable Turn on privileged commands  
exit Exit from the EXEC  
logout Exit from the EXEC  
ping Send echo messages  
resume Resume an active network connection  
show Show running system information  
ssh Open a secure shell client connection  
telnet Open a telnet connection  
terminal Set terminal line parameters  
traceroute Trace route to destination
```

Only a very limited set of informational commands are available in User Exec mode and we won't typically be working here.

Enter the 'show run' command.

```
RouterX>show run  
^  
% Invalid input detected at '^' marker.
```

'show run' is a valid command but it's run at Privileged Exec mode, not User Exec, so the command fails.

This is the most common issue to trip up beginners at the IOS command line. If you see the 'invalid input' error then check you are at the correct level for the command you are trying to run.

Exploring Privileged Exec (Enable) Mode and Context Sensitive Help

Enter Privileged Exec mode. This mode is often commonly known as Enable mode. Notice that the prompt changes to 'Router#'

```
Router>enable
Router#
```

Drop back to User Exec mode.

```
Router#disable
Router>
```

Go back to Privileged Exec mode by using shortened command abbreviation.

```
Router>en
Router#
```

Command abbreviation only works when you enter letters which could only match one unique command. Attempt to return to User Exec mode by entering the command 'di'

```
Router#di
% Ambiguous command:  "di"
```

Check to see all the possible commands which begin with the letters 'di'

```
Router#di?
dir  disable  disconnect
```

We can see that the shortest combination we could use for Disable would be 'disa'

We can access detailed informational and debug output in Privileged Exec mode.

Check to see all commands that begin with 'sh'

```
Router#sh?
Show
```

'show' is the only command that begins with 'sh' so we can use that as the abbreviation.

Enter 'sh ?' to see all available show commands. Notice that we have now included a space before the question mark. This enters context sensitive help for the 'show' command.

```
Router#sh ?
aaa Show AAA values
access-lists List access lists
arp Arp table
cdp CDP information
class-map Show QoS Class Map
clock Display the system clock
controllers Interface controllers status
crypto Encryption module
debugging State of each debugging option
dhcp Dynamic Host Configuration Protocol status
dot11 IEEE 802.11 show information
file Show filesystem information
flash: display information about flash: file system
flow Flow information
frame-relay Frame-Relay information
history Display the session command history
hosts IP domain-name, lookup style, nameservers, and host
table
interfaces Interface status and configuration
ip IP information
ipv6 IPv6 information
license Show license information
line TTY line information
--More--
```

Press the Enter key when you see '—More—' to cycle through the additional output one line at a time.

```
- Output truncated -
history Display the session command history
hosts IP domain-name, lookup style, nameservers, and host
table
interfaces Interface status and configuration
ip IP information
ipv6 IPv6 information
license Show license information
line TTY line information
lldp LLDP information
--More--
```

One line at a time is a very slow way to view additional output so press the Space Bar to cycle through it one page at a time instead.

```
- Output truncated -
sessions Information about Telnet connections
snmp snmp statistics
spanning-tree Spanning tree topology
ssh Status of SSH server connections
standby standby configuration
startup-config Contents of startup configuration
storm-control Show storm control configuration
tcp Status of TCP connections
tech-support Show system information for Tech-Support
--More--
```

Keep hitting the Space Bar until you reach the end of the output.

```
- Output truncated -
standby standby configuration
startup-config Contents of startup configuration
storm-control Show storm control configuration
tcp Status of TCP connections
tech-support Show system information for Tech-Support
terminal Display terminal configuration parameters
users Display information about terminal lines
version System hardware and software status
vlan-switch VTP VLAN status
vtp Configure VLAN database
Router#sh
```

Check the possible options for the 'show aaa' command. (We're using aaa for illustrative purposes here. Don't worry about the meaning of the individual aaa commands, they're not important for this exercise.)

```
Router#sh aaa ?
local Show AAA local method options
sessions Show AAA sessions as seen by AAA Session MIB
user Show users active in AAA subsystem
```

Context sensitive help can be very useful if you're not sure about the exact command you need to use. Unfortunately its use may be disabled in the simulator questions on the CCNA exam so you'll need to actually know the commands.

Enter 'sh aaa us' and then hit the Tab key to see Tab completion in action.

```
Router#sh aaa user
```

The Tab key will complete a partially entered command for you. Again this will only work if you've entered enough letters to be a unique match.

Enter the command 'sh aaa user all'.

```
Router#sh aaa user all
Router#
```

Notice that you do not get any output when you enter the command. This is not an error - AAA has not been configured. The CLI simply returns to the Enable prompt because there is nothing to show.

Enter the command 'sh aaa usor all'

```
Router#sh aaa usor all
                ^
% Invalid input detected at '^' marker.
```

If you enter an illegal command you will get an error message.

Here we made a typo. The CLI warns us that invalid input was detected and shows us the location of the typo is at the 'o' of usor. We typed usor instead of user.

Enter the command 'sh aaa' and hit Enter.

```
Router#sh aaa
% Incomplete command.
```

The router warns us that we've entered an incomplete command, we need to enter additional input. We could enter 'sh aaa ?' again to see the available options.

Explore Global Configuration Mode

Enter Global Configuration mode. (The command can be abbreviated to 'conf t'.)

```
Router#configure terminal
Enter configuration commands, one per line.  End with
CNTL/Z.
Router(config)#
```

Notice that the prompt changes to 'Router(config)#'

Global Configuration mode is where we can enter configuration which affects the device as a whole (as opposed to configuring a particular interface for example).

Add a couple of host entries. (Don't worry what this command does for now, we're going to use it to illustrate command history in a second.)

```
Router(config)#ip host Server1 1.1.1.1
Router(config)#ip host Server2 2.2.2.2
```

Attempt to change the hostname of the device to R1 by entering the command 'R1'.

```
Router(config)#R1
                        ^
% Invalid input detected at '^' marker.
```

Oops we forgot to enter the 'hostname' keyword at the start of the command.

Hit the Up Arrow on your keyboard to cycle back to the previous command.

```
Router(config)#R1
```

Enter Ctrl-A to bring the cursor to the beginning of the line and change the entry to 'hostname R1'. This is quicker than typing the command again. (We can also use 'Ctrl-E' to bring the cursor to the end of the line, and the left and right arrows to move the cursor one character at a time.)

```
Router(config)#hostname R1
R1(config)#
```

Notice that the command prompt changes to show the router's hostname.

Hit the Up Arrow repeatedly to cycle back through your previous command history, and then the Down Arrow to cycle back again. Notice that command

history is specific to your current level in the command hierarchy - only the commands you previously entered in Global Configuration mode are shown.

Enter the command 'show ip interface brief' to check which interfaces are available in the router.

```
R1(config)#show ip interface brief
      ^
% Invalid input detected at '^' marker.
```

You receive the 'invalid input detected' error message but we haven't made a typo. We're getting the error because you have to be at the correct level whenever you enter a command. We're in Global Configuration mode but 'show' commands are run in Privileged Exec mode.

We can override this for 'show' commands by entering 'do' at the start of the command. This works from any level in the command hierarchy.

Enter the correct command to check what interfaces are available from Global Configuration mode.

```
R1(config)#do show ip interface brief
Interface IP-Address OK? Method Status Protocol
GigabitEthernet0/0 unassigned YES NVRAM administratively down down
GigabitEthernet0/1 unassigned YES NVRAM administratively down down
GigabitEthernet0/2 unassigned YES NVRAM administratively down down
Vlan1 unassigned YES NVRAM administratively down down
```

Enter Interface Configuration mode for one of your interfaces.

```
R1(config)#interface gigabitEthernet 0/0
R1(config-if)#
```

Notice that the prompt changes to indicate you are in Interface Configuration mode.

Drop back down to Global Configuration mode.

```
R1(config-if)#exit
R1(config)#
```

The 'exit' command drops back down one level.

Hit the Up Arrow and go back to Interface Configuration mode.

```
R1(config)#interface gigabitEthernet 0/0
R1(config-if)#
```

Drop all the way back down to Privilege Exec mode with a single command.

```
R1(config-if)#end
R1#
```

The 'end' command drops back down to Privilege Exec mode from any level. You can also achieve this by entering 'Ctrl-C'

View the entire device configuration.

```
R1#show running-config
Building configuration...

Current configuration : 737 bytes
!
version 15.1
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
hostname R1
!
Output truncated -
```

View the entire configuration, starting from the hostname.

```
R1#sh run | begin hostname
hostname R1
!
!
!
!
!
!
!
!
no ip cef
no ipv6 cef
!
Output truncated -
```

Note that the IOS command line is not case sensitive, except when we pipe commands.

```
R1#sh run | begin Hostname
R1#
```

Here we entered 'Hostname' with a capital letter at the start, but this is not how it is shown in the configuration. The router could find no instance of 'Hostname' so it returns no output.

View configuration lines which include the word 'interface'.

```
R1#show run | include interface
interface GigabitEthernet0/0
interface GigabitEthernet0/1
interface GigabitEthernet0/2
interface Vlan1
```

View all configuration lines which do not include the word 'interface'.

```
R1#show run | exclude interface
Building configuration...
```

```
Current configuration : 737 bytes
!
version 15.1
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
hostname R1
!
Output truncated -
```

IOS Configuration Management

Copy the running configuration to the startup configuration.

```
R1#copy run start
Destination filename [startup-config]?
Building configuration...
[OK]
```

Change the hostname of the router to RouterX

```
R1#config t
Enter configuration commands, one per line. End with
CNTL/Z.
R1(config)#hostname RouterX
RouterX(config)#
```

Notice that when you enter a command in IOS it takes effect immediately, we can see the command prompt changes to show the new hostname.

Check what hostname will be used when the system reboots.

```
RouterX(config)#do show startup-config
Using 737 bytes
!
version 15.1
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
hostname R1
!
```

Commands take effect immediately but are not persistent across a reboot until we save them.

Save the current running configuration so it will be applied next time the router is reloaded.

```
RouterX#copy run start
Destination filename [startup-config]?
Building configuration...

[OK]
```

Verify the new hostname will be applied following a reboot.

```
RouterX#show start
Using 742 bytes
!
version 15.1
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
hostname RouterX
!
```

Backup the current running configuration to flash memory in the router.

```
RouterX#copy run flash:
Destination filename [running-config]? config-backup
Building configuration...
[OK]
```

It's not a good idea to back up a device to the device itself, so enter the command to back the running configuration up to an external TFTP server.

```
RouterX#copy run tftp
Address or name of remote host []? 10.10.10.10
Destination filename [RouterX-config]?

Writing running-config.....
%Error opening tftp://10.10.10.10/RouterX-config (Timed out)
```

(The command will try to run for a while and then time out and fail because we didn't set up connectivity to a TFTP server in the lab.)

Reload the device and check it comes back up with the expected configuration with hostname RouterX.

```
RouterX#reload
Proceed with reload? [confirm]
```