

## Cisco Basics - Privileged Mode

### Introduction

To get into Privileged Mode we enter the "*Enable*" command from User Exec Mode. If set, the router will prompt you for a password. Once in Privileged Mode, you will notice the prompt changes from ">" to a "#" to indicate that we are now in Privileged Mode.

The Privileged Mode (and Global Configuration Mode ) is used mainly to configure the router, enable interfaces, setup security, define dialup interfaces etc.

I have put a screen shot of the router to give you an idea of the commands available in Privileged Mode in comparison to the User Exec Mode. Remember that these commands have sub-commands and can get quite complicated:

```
Cisco Router - Telnet - SecureCRT
File Edit View Options Transfer Script Window Help
swiftpond#?
Exec commands:
<1-99>          Session number to resume
access-enable   Create a temporary Access-List entry
access-profile  Apply user-profile to interface
access-template Create a temporary Access-List entry
archive         Manage archive files
bfe             For manual emergency modes setting
cd              Change current directory
clear           Reset functions
clock           Manage the system clock
configure       Enter configuration mode
connect         Open a terminal connection
copy            Copy from one file to another
debug           Debugging functions (see also 'undebug')
delete          Delete a file
dir             List files on a filesystem
disable         Turn off privileged commands
disconnect      Disconnect an existing network connection
enable          Turn on privileged commands
erase           Erase a filesystem
exit            Exit from the EXEC
help            Description of the interactive help system
isdn            Make/disconnect an isdn data call on a BRI interface
lock            Lock the terminal
login           Log in as a particular user
logout          Exit from the EXEC
more            Display the contents of a file
name-connection Name an existing network connection
no              Disable debugging functions
pad             Open a X.29 PAD connection
ping            Send echo messages
ppp             Start IETF Point-to-Point Protocol (PPP)
pwd             Display current working directory
reload          Halt and perform a cold restart
resume          Resume an active network connection
rlogin          Open an rlogin connection
rsh             Execute a remote command
send            Send a message to other tty lines
set             Set system parameter (not config)
setup           Run the SETUP command facility
show            Show running system information
slip            Start Serial-line IP (SLIP)
start-chat      Start a chat-script on a line
sysstat         Display information about terminal lines
telnet          Open a telnet connection
terminal        Set terminal line parameters
test            Test subsystems, memory, and interfaces
traceroute      Trace route to destination
tunnel          Open a tunnel connection
undebg          Disable debugging functions (see also 'debug')
undelete        Undelete a file
verify          Verify a file
where           List active connections
write           Write running configuration to memory, network, or terminal
x28             Become an X.28 PAD
x3             Set X.3 parameters on PAD

swiftpond#
swiftpond#
```

As you can see, there is a wider choice of commands in Privileged Mode.

Now, when you want to configure certain services or parts of the router you will need to enter Global Configuration Mode from within Privileged Mode. If you're confused by now with the different modes available try to see it this way :

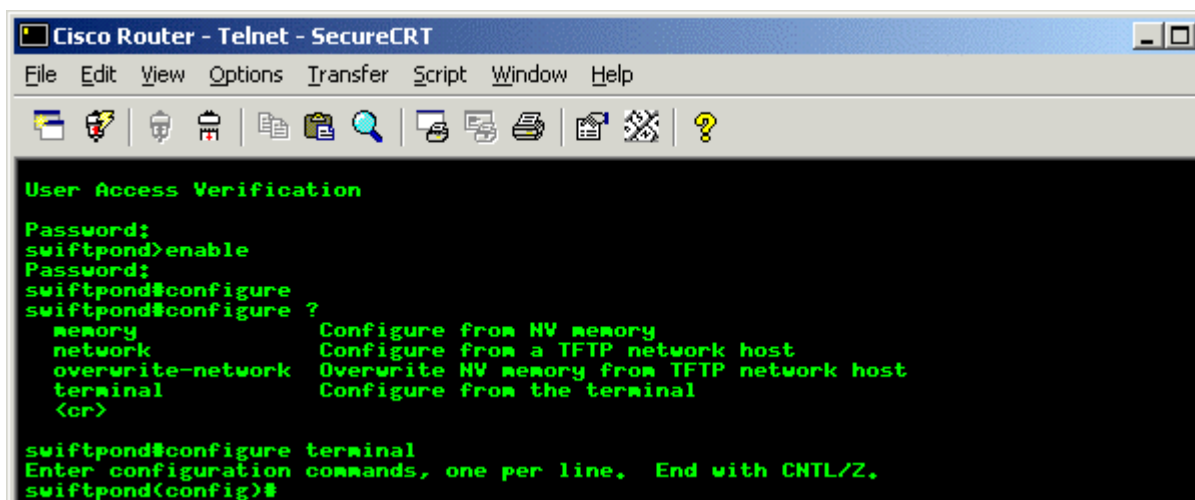
User Exec Mode (distinguished by the ">" prompt) is your first mode, which is used to get statistics from router, see which version IOS you're running, check memory resources and a few more things.

Privileged Mode (distinguished by the "#" prompt) is the second mode. Here you can enable or disable interfaces on the router, get more detailed information on the router, for example, view the running configuration of the router, copy the configuration, load a new configuration to the router, backup or delete the configuration, backup or delete the IOS and a lot more.

Global Configuration Mode (distinguished by the "(config)#" prompt) is accessible via Privileged Mode. In this mode you're able to configure each interface individually, setup banners and passwords, enable secrets (encrypted passwords), enable and configure routing protocols and a lot more. I dare say that 70% of the time you want to configure or change something on the router, you will need to be in this mode.

## Getting into Global Configuration

The picture below shows you how to enter Global Configuration Mode:



```
Cisco Router - Telnet - SecureCRT
File Edit View Options Transfer Script Window Help
User Access Verification
Password:
swiftpond>enable
Password:
swiftpond#configure
swiftpond#configure ?
  memory          Configure from NV memory
  network         Configure from a TFTP network host
  overwrite-network Overwrite NV memory from TFTP network host
  terminal        Configure from the terminal
  <cr>

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

As you can see, I have telneted into the router and it prompted me for a password. I entered the password, which is not shown, at this point I am in User Exec Mode and then entered "enable" in order to get into the Privileged Mode. From here to get into Global Configuration Mode you need to enter the "configure *selection*" command.

Now you must be wondering what the various parameters shown in the picture are, under the "configure" command. These allow you to select how you will configure the router:

- *Configure Memory* means you enter Global Configuration Mode and are configuring the router in its NVRAM. This command will force the router to load up the startup-config file stored in the NVRAM and then you can proceed with

the configuration. When you're happy with the configuration, save it to NVRAM by entering "*copy running-config startup-config*".

- *Configure Network* means you enter Global Configuration Mode and load a startup-config file from a remote router (using tftp) into your local router's memory and configure it. Once you're finished, you need to enter "*copy running-config tftp*" which will force the router to copy its memory configuration onto a tftp server. The router will prompt you for the IP address of the remote tftp server.
- *Configure Overwrite-network* means that you overwrite the NVRAM's configuration with a configuration stored on a tftp server. Issuing this command will force the router to prompt for an IP address of the remote tftp server. Personally, I have never needed to use this command.
- *Configure Terminal* means you enter Global Configuration Mode and work with the configuration which is already loaded into the router's memory (Cisco calls this the *running-config*). This is the most popular command, as in most cases you need to modify or re-configure the router on the spot and then save your changes.

You will need to save this configuration otherwise everything you configure will be lost upon power failure or reboot of the router !

Below are the commands you need to enter to save the configuration, depending on your network setup:

- *Copy running-config startup-config*: Copies the configuration which is running in the router's RAM in to the NVRAM and gives it a file name of startup-config (default). If one already exists in the NVRAM, it will be overwritten by the new one.
- *Copy running-config tftp*: Copies the configuration which is running in the router's RAM in to a tftp server which might be running on your network. You will be asked for the IP address of the tftp server and given the choice to select a filename for the configuration. Some advanced routers can also act as tftp servers.

## **Generic Configuration**

There are a few standard things with which you always need to configure the router . For example, a hostname. This is also used as a login name for the remote router to which your router needs to authenticate. Before we get stuck into the interface configuration we are going to run through a few of these commands. The following examples assume no passwords have been set as yet and that the router has a default hostname of "router":

We connect to the router via the console port using the serial cable and type the following

Router> enable (gets us into Privileged Mode)

Router# configure terminal (This command gets us into the appropriate Global Configuration Mode as outlined above)

Router(config)# hostname swiftpond (This command sets the router's hostname to swiftpond. From this moment onwards, swiftpond will appear before the ">" or "#" depending on which mode we are in)

swiftpond(config)# username router2.isp password firewallcx (Here we are telling the router that the remote router which we are connecting to, has a username of "router2.isp" and our password to authenticate to router2.isp is "firewallcx")

This is a standard way of authentication with Cisco routers. Your router's hostname is your login name and your password (in our case "firewallcx") is entered at the same time you define the remote router's hostname.

Next we create a static route so the router will pass all packets originating from our network to the remote router. This is usually the case when you connect to your isp.

swiftpond(config)# ip route 0.0.0.0 0.0.0.0 139.130.34.43 (Here we tell our router to create a default route where any packet -defined by the first 0.0.0.0- no matter what subnetmask -defined by the second 0.0.0.0- is to be sent to ip 139.130.34.43 which would be the router we are connecting to)

In the case where you were not configuring the router to connect to the Internet but to join a small WAN which connects a few offices, then you probably want to use a routing protocol:

swiftpond(config)# router rip (Enables RIP routing protocol. After this command you enter the routing protocols configuration section -see below- where you can change timing parameters and other)

swiftpond(config-router)#

At this prompt you can fine tune RIP or just leave it to the default setting which will work fine. The "exit" command takes you one step back:

swiftpond(config-router)# exit

swiftpond(config)#

Alternatively, you can use IGRP as a routing protocol, in which case you would have to enter the following:

```
swiftpond(config)# router igrp 1 (The "1" defines the Autonomous system number)
```

```
swiftpond(config-router)#
```

Again, the "exit" command will take you back one step:

```
swiftpond(config-router)# exit
```

```
swiftpond(config)#
```

After that, we need to create a dialer list which our WAN interface BRI (ISDN) will use to make a call to our ISP.

```
swiftpond(config)# dialer-list 1 protocol ip permit (Now we are telling the router to create a dialer list and bind it to group 1. The "protocol ip permit" tells the router to initiate a call for an ip packet)
```

I'll give you a quick example to make sure you understand the reason we put this command:

If you launched your web browser, it would send an http request to the server you have set as a homepage e.g [www.firewall.cx](http://www.firewall.cx). This request which your computer is going to send, is encapsulated in an ip packet that will cause your router to initiate a connection, as it is now configured to do so.

The dialup interface for Cisco routers is broken into 2 parts: a Dialer-list and a Dialer-group.

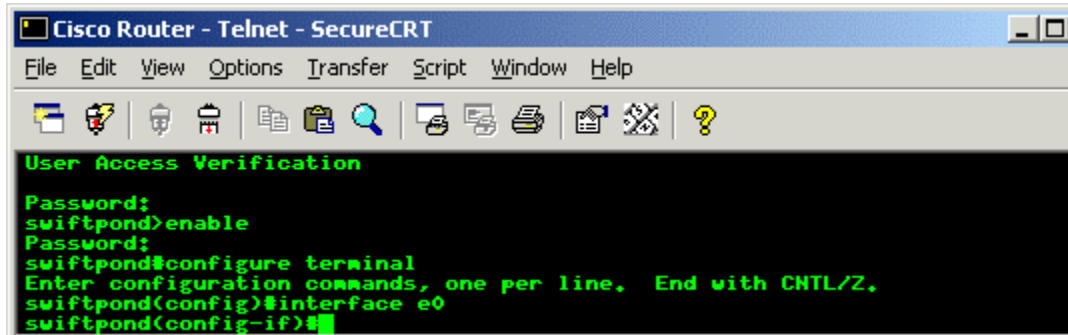
The Dialer-list defines the rules for placing a call. Later on when you configure the WAN interface, you bind that Dialer-list to the interface by using the Dialer-group command (shown later on).

## **Configuring Interfaces**

In our example we said we have a router with one Ethernet and one basic ISDN interface (max of 128Kbit). We are going to go through the process of configuring the interfaces. We will start with the Ethernet Interface.

In order to configure the interface, we need to be in Global Configuration Mode, so we need to type first "enable" in order to get into Privileged Mode and then "configure terminal" to get into the appropriate Global Configuration Mode (as explained above). Now we need to select the interface we want to configure, in this case the first ethernet interface (E0) so we type "interface e0".

This picture shows clearly all the steps:



```
Cisco Router - Telnet - SecureCRT
File Edit View Options Transfer Script Window Help
User Access Verification
Password:
swiftpond>enable
Password:
swiftpond#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
swiftpond(config)#interface e0
swiftpond(config-if)#
```

Any commands entered here will affect the first ethernet interface only. So we start with the IP address. It's important to understand that this IP address would be visible to both networks to which the router is connected. If we were connecting to the Internet then everyone would be able to see this IP. Furthermore, the IP address would also be the default gateway for our firewall or machine which would physically connect directly to the router.

The following commands will configure the ethernet interface's IP address::

```
(config-if)# ip address 192.168.0.1 255.255.255.0
```

or

```
(config-if)# ip address 139.130.4.5 255.255.255.0 secondary
```

Now that we have given e0 its IP address, we need to give the ISDN interface its IP as well, so we need to move to the correct interface by typing the following:

```
(config-if)# exit (this exits from the e0 interface configuration)
```

```
(config-if)# interface bri0 (this command enters the configuration for the first ISDN interface)
```

```
(config-if)# ip address 10.0.0.2 255.255.255.224 (this command sets the IP address for BRI 0 which is also known as the WAN IP address)
```

Now when it comes to configuring WAN interfaces, you need more than just an IP address (LAN interfaces such as E0 are a lot easier to configure). You need to set the encapsulation type, the authentication protocol the router will use to authenticate to the remote router, the phone number it will need to dial and a few more:

```
(config-if)# encapsulation ppp (This command sets the packet's encapsulation to ppp which is 100% compatible with all routers no matter what brand)
```

`(config-if)# dialer string 0294883452` (This command tells the router which phone number it needs to dial in order to establish a connection with our remote router e.g your ISP)

`(config-if)# dialer group 1` (This command tells the router to use the dialer list 1 (configured previously) to initiate a connection)

`(config-if)# idle-timeout 2000000` (This command is optional and allows us to set an idle timeout so if the router is idle for so many seconds, it will disconnect. A value of 2 million seconds means the router will never disconnect)

`(config-if)# isdn switch-type basic-net3` (This command tells the router the type of ISDN interface we are using. Each country has its own type, so you need to consult your Cisco manual to figure out which type you need to put here)

`(config-if)# dialer load-threshold 125 outbound` (This command is optional and allows us to specify a threshold upon which it will place another call. The value it takes is from 1 to 255. A value of 125 means bring up the second B channel if either the inbound or outbound traffic load is 50%.

That pretty much does it for our ISDN (WAN) interface. All you need to do now is to SAVE the configuration !

Well I hope it wasn't too bad for you, since there is a quite a bit of information on this page. I encourage you to read through it again until you understand what is going on, then you will find it a breeze to configure a Cisco router yourself !