

Cisco Router Configuration Tutorial

Randy Ong
ran@hiddenlinux.com

This document covers basic Cisco router IP configuration using the command-line interface

Acknowledgments

The following sources were extremely useful:
Leinwand, Pinsky, and Culpepper. *Cisco Router Configuration*. Indianapolis, Indiana: Cisco Press, 1998.
Cisco Systems, Inc., <http://www.cisco.com>

Conventions

Important terms and concepts, when they are introduced, may be displayed in bold. Commands included in the body of the text will be displayed in `this font`. All names and addresses used in examples are just that, examples, and should not be used on your network. Do not type them in verbatim when configuring your system. Finally, in some examples where the command requires an IP address as an argument, the IP address may be represented in this way, xx.xx.xx.xx, or aa.bb.cc.dd. You will never actually use these strings when configuring your system. They are merely a convention of this document to specify that you should substitute the appropriate IP address at that point.

1. What this document covers

There are several methods available for configuring Cisco routers. It can be done over the network from a TFTP server. It can be done through the menu interface provided at bootup, and it can be done from the menu interface provided by using the command `setup`. It can also be done from a configuration saved to memory. This tutorial does not cover these methods. It covers configuration from the IOS command-line interface only.

Note : that this tutorial **does not** cover physically connecting the router to the networks it will be routing for. It covers operating system configuration only.

1.1 Reasons for using the command-line

There are two main reasons for using the command-line interface instead of a menu driven interface. One is speed. Once you have invested the time to learn the command-line commands, you can perform many operations much more quickly than by using a menu. This is basically true of all command-line vs. menu interfaces. What makes it especially efficient to learn the command-line interface of the Cisco IOS is that it is standard across all Cisco routers.

Secondly, there is the ability to configure individual interfaces without interrupting service on other interfaces. By definition, routers have multiple interfaces. Routers such as those in the Cisco 7200 series have several hot-swappable modules of multiple ports each. It is a valuable skill to be able to configure these individually from the command-line.

1.2 Document structure

The first part of this document will introduce the command modes of the IOS, and the commands necessary to do a basic configuration of a Cisco router. The second part of the document will demonstrate the use of these commands in a case study. The case study is an actual configuration completed by the author of this document.

2. Getting started

Initially you will probably configure your router from a terminal. If the router is already configured and at least one port is configured with an IP address, and it has a physical connection to the network, you might be able to `telnet` to the router and configure it across the network. If it is not already configured, then you will have to directly connect to it with a terminal and a serial cable. With any Windows box you can use HyperTerminal to easily connect to the router. Plug a serial cable into a serial (COM) port on the PC and the other end into the console port on the Cisco router. Start HyperTerminal, tell it which COM port to use and click `OK`. Set the speed of the connection to `9600 baud` and click `OK`. If the router is not on, turn it on.

If you wish to configure the router from a **Linux** box, either **Seyon** or **Minicom** should work. At least one of them, and maybe both, will come with your Linux distribution. Often you will need to hit the **Enter** key to see the prompt from the router. If it is unconfigured it will look like this:

```
Router>
```

If it has been previously configured with a hostname, it will look like this:

```
hostname of router>
```

If you have just turned on the router, after it boots it will ask you if you wish to begin initial configuration. **Say no**. If you say `yes`, it will put you in the menu interface. Say `no`.

2.1 Modes

The Cisco IOS command-line interface is organized around the idea of **modes**. You move in and out of several different modes while configuring a router, and which mode you are in determines what commands you can use. Each mode has a set of commands available in that mode, and some of these commands are *only* available in that mode. In any mode, typing a question mark will display a list of the commands available in that mode.

```
Router>?
```

2.2 Unprivileged and privileged modes

When you first connect to the router and provide the password (if necessary), you enter EXEC mode, the first mode in which you can issue commands from the command-line. From here you can use such **unprivileged** commands as **ping**, **telnet**, and **rlogin**. You can also use some of the **show** commands to obtain information about the system. In unprivileged mode you use commands like, **show version** to display the version of the IOS the router is running. Typing **show ?** will display all the **show** commands available in the mode you are presently in.

```
Router>show ?
```

You must enter privileged mode to configure the router. You do this by using the command **enable**. Privileged mode will usually be password protected unless the router is unconfigured. You have the option of not password protecting privileged mode, but it is HIGHLY recommended that you do. When you issue the command **enable** and provide the password, you will enter privileged mode.

To help the user keep track of what mode they are in, the command-line prompt changes each time you enter a different mode. When you switch from unprivileged mode to privileged mode, the prompt changes from:

```
Router>
to
Router#
```

This would probably not be a big deal if there were just two modes. There are, in fact, numerous modes, and this feature is probably indispensable. Pay close attention to the prompt at all times.

Within privileged mode there are many **sub-modes**. In this document I do not closely follow Cisco terminology for this hierarchy of modes. I think that my explanation is clearer, frankly. Cisco describes two modes, unprivileged and privileged, and then a hierarchy of commands used in privileged mode. I reason that it is much clearer to understand if you just consider there to be many sub-modes of privileged mode, which I will also call **parent** mode. Once you enter privileged mode (parent mode) the prompt ends with a pound sign (#). There are numerous modes you can enter only after entering privileged mode. Each of these modes has a prompt of the form:

```
Router(arguments)#
```

They still all end with the pound sign. They are subsumed within privileged mode. Many of these modes have sub-modes of their own. Once you enter privileged mode, you have access to all the configuration information and options the IOS provides, either directly from the parent mode, or from one of its submodes.

3. Configuring

If you have just turned on the router, it will be completely unconfigured. If it is already configured, you may want to view its current configuration. Even if it has not been previously

configured, you should familiarize yourself with the **show** commands before beginning to configure the router. Enter privileged mode by issuing the command **enable**, then issue several **show** commands to see what they display. Remember, the command **show ?** will display all the **show** commands available in the current mode. Definitely try out the following commands:

```
Router# show interfaces
Router# show ip protocols
Router# show ip route
Router# show ip arp
```

When you enter privileged mode by using the command **enable**, you are in the top-level mode of privileged mode, also known in this document as "parent mode." It is in this top-level or parent mode that you can display most of the information about the router. As you now know, you do this with the **show** commands. Here you can learn the configuration of interfaces and whether they are up or down. You can display what IP protocols are in use, such as dynamic routing protocols. You can view the route and ARP tables, and these are just a few of the more important options.

As you configure the router, you will enter various sub-modes to set options, then return to the parent mode to display the results of your commands. You also return to the parent mode to enter other sub-modes. To return to the parent mode, you hit **ctrl-z**. This puts any commands you have just issued into affect, and returns you to parent mode.

3.1 Global configuration (config)

To configure any feature of the router, you must enter configuration mode. This is the first sub-mode of the parent mode. In the parent mode, you issue the command **config**.

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

As demonstrated above, the prompt changes to indicate the mode that you are now in.

In configuration mode you can set options that apply system-wide, also referred to as "global configurations." For instance, it is a good idea to name your router so that you can easily identify it. You do this in configuration mode with the **hostname** command.

```
Router(config)# hostname Hidden
Hidden(config)#
```

As demonstrated above, when you set the name of the host with the **hostname** command, the prompt immediately changes by replacing **Router** with **Hidden**. (**Note:** It is a good idea to name your routers with an organized naming scheme.)

Another useful command issued from config mode is the command to designate the DNS server to be used by the router:

```
Hidden(config)# ip name-server aa.bb.cc.dd
Hidden(config)# ctrl-Z
Hidden #
```

This is also where you set the password for privileged mode.

```
Hidden (config)#enable secret examplepassword
Hidden (config)#ctrl-Z
Hidden #
```

Until you hit **ctrl-Z** (or type **exit** until you reach parent mode) your command has not been put into affect. You can enter config mode, issue several different commands, then hit **ctrl-Z** to activate them all. Each time you hit **ctrl-Z** you return to parent mode and the prompt:

```
Hidden #
```

Here you use **show** commands to verify the results of the commands you issued in config mode. To verify the results of the **ip name-server** command, issue the command **show host**.

3.2 Configuring interfaces

Cisco interface naming is straightforward. Individual interfaces are referred to by this convention:

```
media type slot#/port#
```

"Media type" refers to the type of media that the port is an interface for, such as Ethernet, Token Ring, FDDI, serial, etc. Slot numbers are only applicable for routers that provide slots into which you can install modules. These modules contain several ports for a given media. The 7200 series is an example. These modules are even hot swappable. You can remove a module from a slot and replace it with a different module, without interrupting service provided by the other modules installed in the router. These slots are numbered on the router.

Port number refers to the port in reference to the other ports in that module. Numbering is left-to-right, and all numbering starts at 0, not at one. For example, a Cisco 7206 is a 7200 series router with six slots. To refer to an interface that is the third port of an Ethernet module installed in the sixth slot, it would be interface Ethernet 6/2. Therefore, to display the configuration of that interface you use the command:

```
Hidden # show interface ethernet 6/2
```

If your router does not have slots, like a 1600, then the interface name consists only of:

```
media type port#
```

For example:

```
Hidden #show interface serial 0
```

Here is an example of configuring a serial port with an IP address:

```
Hidden# config
Hidden(config)# interface serial 1/1
Hidden(config-if)# ip address 192.168.155.2 255.255.255.0
Hidden(config-if)# no shutdown
Hidden(config-if)# ctrl-Z
Hidden#
```

Then to verify configuration:

```
Hidden# show interface serial 1/1
```

Note : the **no shutdown** command. An interface may be correctly configured and physically connected, yet be "administratively down." In this state it will not function. The command for causing an interface to be administratively down is **shutdown**.

```
Hidden(config)# interface serial 1/1
Hidden(config-if)# shutdown
Hidden(config-if)# ctrl-Z
Hidden# show interface serial 1/1
```

In the Cisco IOS, the way to reverse or delete the results of any command is to simply put **no** in front of it. For instance, if we wanted to unassign the IP address we had assigned to interface serial 1/1:

```
Hidden(config)# interface serial 1/1
Hidden(config-if)# no ip address 192.168.155.2 255.255.255.0
Hidden(config-if)# ctrl-Z
Hidden# show interface serial 1/1
```

Configuring most interfaces for LAN connections might consist only of assigning a network layer address and making sure the interface is not administratively shutdown. It is usually not necessary to stipulate data-link layer encapsulation. **Note** that it is often necessary to stipulate the appropriate data-link layer encapsulation for WAN connections, such as frame-relay and ATM. Serial interfaces default to using HDLC. A discussion of data-link protocols is outside the scope of this document. You will need to look up the IOS command **encapsulation** for more details.

3.3 Routing

IP routing is automatically enabled on Cisco routers. If it has been previously disabled on your router, you turn it back on in config mode with the command **ip routing**.

```
Hidden(config)# ip routing
Hidden(config)# ctrl-Z
```

There are two main ways a router knows where to send packets. The administrator can assign **static routes**, or the router can learn routes by employing a **dynamic routing protocol**.

These days static routes are generally used in very simple networks or in particular cases that necessitate their use. To create a static route, the administrator tells the router operating system that any network traffic destined for a specified network layer address should be forwarded to a similarly specified network layer address. In the Cisco IOS this is done with the **ip route** command.

```
Hidden# config
Hidden(config)# ip route 172.16.0.0 255.255.255.0 192.168.150.1
Hidden(config)# ctrl-Z
Hidden# show ip route
```

Two things to be said about this example. First, the packet destination address must include the subnet mask for that destination network. Second, the address it is to be forwarded to is the specified address of the next router along the path to the destination. This is the most common way of setting up a static route, and the only one this document covers. Be aware, however, that there are other methods.

Dynamic routing protocols, running on connected routers, enable those routers to share routing information. This enables routers to *learn* the routes available to them. The advantage of this method is that routers are able to adjust to changes in network topologies. If a route is physically removed, or a neighbor router goes down, the routing protocol searches for a new route. Routing protocols can even dynamically choose between possible routes based on variables such as network congestion or network reliability.

There are many different routing protocols, and they all use different variables, known as "metrics," to decide upon appropriate routes. Unfortunately, a router needs to be running the same routing protocols as its neighbors. Many routers can, however, run multiple protocols. Also, many protocols are designed to be able to pass routing information to other routing protocols. This is called "redistribution." The author has no experience with trying to make redistribution work. There is an IOS **redistribute** command you can research if you think this is something you need. This document's companion case study describes an alternative method to deal with different routing protocols in some circumstances. Routing protocols are a complex topic and this document contains only this superficial description of them. There is much to learn about them, and there are many sources of information about them available. An excellent source of information on this topic is Cisco's website, <http://www.cisco.com>.

This document describes how to configure the Routing Information Protocol (RIP) on Cisco routers. From the command-line, we must explicitly tell the router which protocol to use, and what networks the protocol will route for.

```
Hidden# config
Hidden(config)# router rip
Hidden(config-router)# network aa.bb.cc.dd
Hidden(config-router)# network ee.ff.gg.hh
Hidden(config-router)# ctrl-Z
Hidden# show ip protocols
```

Now when you issue the **show ip protocols** command, you should see an entry describing RIP configuration.

3.4 Saving your configuration

Once you have configured routing on the router, and you have configured individual interfaces, your router should be capable of routing traffic. Give it a few moments to talk to its neighbors, then issue the commands **show ip route** and **show ip arp**. There should now be entries in these tables learned from the routing protocol.

If you turned the router off right now, and turned it on again, you would have to start configuration over again. Your **running configuration** is not saved to any permanent storage media. You can see this configuration with the command **show running-config**.

```
Hidden # show running-config
```

You **do** want to save your successful running configuration. Issue the command **copy running-config startup-config**.

```
Hidden # copy running-config startup-config
```

Your configuration is now saved to **non-volatile RAM** (NVRAM). Issue the command **show startup-config**.

```
Hidden # show startup-config
```

Now any time you need to return your router to that configuration, issue the command **copy startup-config running-config**.

```
Hidden # copy startup-config running-config
```

3.5 Example configuration

```
Router> enable
Router# config
Router(config)# hostname cisco-7206
cisco-7206(config)# interface serial 1/1
cisco-7206(config-if)# ip address 192.168.155.2 255.255.255.0
cisco-7206(config-if)# no shutdown
cisco-7206(config-if)# ctrl-z
cisco-7206# show interface serial 1/1
cisco-7206# config
cisco-7206(config)# interface ethernet 2/3
cisco-7206(config-if)# ip address 192.168.150.90 255.255.255.0
cisco-7206(config-if)# no shutdown
cisco-7206(config-if)# ctrl-z
cisco-7206# show interface ethernet 2/3
cisco-7206# config
cisco-7206(config)# router rip
cisco-7206(config-router)# network 192.168.155.0
```

```

cisco-7206(config-router)# network 192.168.150.0
cisco-7206(config-router)# ctrl-z
cisco-7206# show ip protocols
cisco-7206# ping 192.168.150.1
cisco-7206# config
cisco-7206(config)# ip name-server 203.176.28.130
cisco-7206(config)# ctrl-z
cisco-7206# ping solaris.ph.psi.net
cisco-7206# config
cisco-7206(config)# enable secret password
cisco-7206(config)# ctrl-z
cisco-7206# copy running-config startup-config
cisco-7206# exit

```

4. Troubleshooting

Inevitably, there will be problems. Usually, it will come in the form of a user notifying you that they cannot reach a certain destination, or any destination at all. You will need to be able to check how the router is attempting to route traffic, and you must be able to track down the point of failure.

You are already familiar with the **show** commands, both specific commands and how to learn what other **show** commands are available. Some of the most basic, most useful commands you will use for troubleshooting are:

```

ExampleName# show interfaces
ExampleName# show ip protocols
ExampleName# show ip route
ExampleName# show ip arp

```

4.1 Testing connectivity

It is very possible that the point of failure is not in your router configuration, or at your router at all. If you examine your router's configuration and operation and everything looks good, the problem might be farther up the line. In fact, it may be the line itself, or it could be another router, which may or may not be under your administration.

One extremely useful and simple diagnostic tool is the **ping** command. Ping is an implementation of the IP Message Control Protocol (ICMP). Ping sends an ICMP echo request to a destination IP address. If the destination machine receives the request, it responds with an ICMP echo response. This is a very simple exchange that consists of:

Hello, are you alive?
Yes, I am.

```
ExampleName# ping xx.xx.xx.xx
```

If the ping test is successful, you know that the destination you are having difficulty reaching is alive and physically reachable.

If there are routers between your router and the destination you are having difficulty reaching, the problem might be at one of the other routers. Even if you **ping** a router and it responds, it might have other interfaces that are down, its routing table may be corrupted, or any number of other problems may exist.

To see where packets that leave your router for a particular destination go, and how far, use the **trace** command.

```
ExampleName# trace xx.xx.xx.xx
```

It may take a few minutes for this utility to finish, so give it some time. It will display a list of all the hops it makes on the way to the destination.

4.2 debug commands

There are several **debug** commands provided by the IOS. These commands are not covered here. Refer to the Cisco website for more information.

4.3 Hardware and physical connections

Do not overlook the possibility that the point of failure is a hardware or physical connection failure. Any number of things can go wrong, from board failures to cut cables to power failures. This document will not describe troubleshooting these problems, except for these simple things.

Check to see that the router is turned on. Also make sure that no cables are loose or damaged. Finally, make sure cables are plugged into the correct ports. Beyond this simple advice you will need to check other sources.

4.4 Out of your control

If the point of failure is farther up the line, the problem might lie with equipment not under your administration. Your only option might be to contact the equipment's administrator, notify them of your problem, and ask them for help. It is in your interest to be courteous and respectful. The other administrator has their own problems, their own workload and their own priorities. Their agenda might even directly conflict with yours, such as their intention to change dynamic routing protocols, etc. You must work with them, even if the situation is frustrating. Alienating someone with the power to block important routes to your network is not a good idea.

For more information on Cisco products, and many documents on networking in general: <http://www.cisco.com/>

ran@hiddenlinux.com