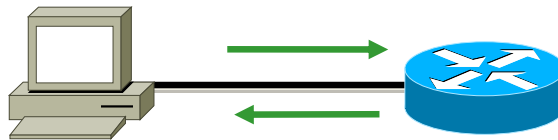


Backing up the configuration

Copy the configuration to a TFTP host and back

Router# **copy running-config tftp**

Router# **copy tftp running-config**



To copy the router's configuration from a router to a TFTP host, you can use either the "**copy running-config tftp**" or the "**copy startup-config tftp**" command. Either one will back up the router configuration that's currently running in DRAM, or that's stored in NVRAM. Note: In order to save off the most current configuration, make sure the startup file matches the running configuration if you plan on utilizing the "**copy startup-config tftp**" command.

If you've changed your router's running-config and want to restore the configuration to the version in startup-config, the easiest way to do this is to use the "**copy startup-config running-config**" command ("**copy start run**" for short).

Note: When you copy or paste a configuration into RAM, the interfaces are shutdown by default. This is especially important if you are configuring the router for the first time, and will be shipping it out to a location where you will not have access to it unless the interface is up. To prevent this, insert "**no shutdown**" commands under each interface needed to at least obtain access to the device.

Fallback

The following commands can be utilised to have a router boot an IOS image from another source:

```
Router# config t
Router(config)# boot system flash ios_filename
Router(config)# boot system tftp ios_filename tftp_address
Router(config)# boot system rom
```

Note: Flash, TFTP server, ROM will be attempted in that order

Cisco routers, by default, load the IOS from Flash memory. However, what happens if the flash memory fails or the file in flash memory becomes corrupted?

By default, the Cisco routers will look for a TFTP server to load an IOS from, and if that fails, some routers, depending on the model, will load a mini-ios from ROM so that an IOS can be restored into flash memory.

Command syntax and parameter descriptions:

boot system flash [*flash-fs:*] [*partition-number:*] [*filename*]

- *flash-fs:* (Optional) Flash file system containing the system image to load at startup. The colon is required.
- *partition-number:* (Optional) Number of the flash memory partition that contains the system image to boot, specified by the optional *filename* argument. If you do not specify a filename, the router loads the first valid file in the specified partition of flash memory. This argument is only valid on routers that can be partitioned.
- *filename* (Optional when used with the **boot system flash** command) Name of the system image to load at startup. This argument is case sensitive. If you do not specify a *filename*, the router loads the first valid file.

Command syntax is similar for **boot system tftp** and **boot system rom** commands.

ROM Monitor Mode

- If the IOS in Flash is corrupt or missing and no network connectivity is available, and the default fallback procedure fails:
 - The router will enter ROM monitor mode

```
System Bootstrap, Version 11.3(2)XA4, RELEASE SOFTWARE
(fc1)
Copyright (c) 1999 by cisco Systems, Inc.
TAC:Home:SW:IOS:Specials for info
C2600 platform with 65536 Kbytes of main memory
rommon 1 >
rommon 2 > confreg 0x2142
You must reset or power cycle for new config to take effect
rommon 3 > i
```

**Remember: when you boot your router and see “rommon” this is bad!
You’re IOS in flash is missing or corrupt.**

In the above example, the router was rebooted and the ctrl-break key stroke was pressed, which took the router into ROM monitor mode.

You would do this to provide password recovery by changing the configuration register to 0x2142, as shown above.

When you have completed the password recovery, set the configuration register back to 0x2102 for normal operation.

The default for a router is to look in flash memory for the IOS, NVRAM for the startup-config

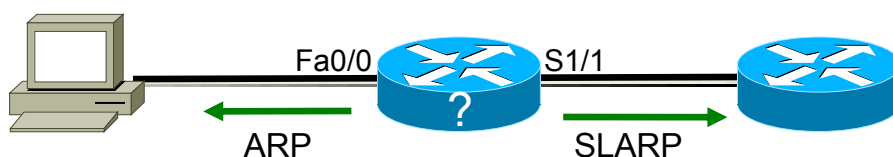
If this fails, the default is to look in flash, then look for a TFTP server on a network, then run a mini-ios from ROM.

If all this fails, then the router will load ROM monitor mode.

Auto-Install

Issue the following commands to stop a router from attempting to pull a configuration from another router or from a network host:

```
Router(config)# no service config  
Router(config)# no boot network
```



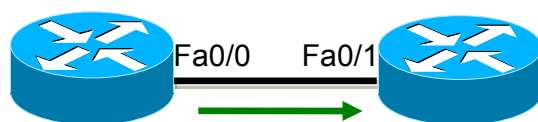
The auto-install “feature” is annoying at best. If a router is powered up, has no configuration and sees Carrier Detect on an interface, it will look for an IP address by using ARP on a LAN and/or SLARP (Serial Line ARP) on a serial interface. You can disable this feature with the “**no service config**” command and the “**no boot network**” command from global configuration mode.

Making Your Router a TFTP Server

Issue the following commands to make your router a TFTP server

```
Router# config t
```

```
Router(config)# tftp-server flash: [press tab or ?]
```

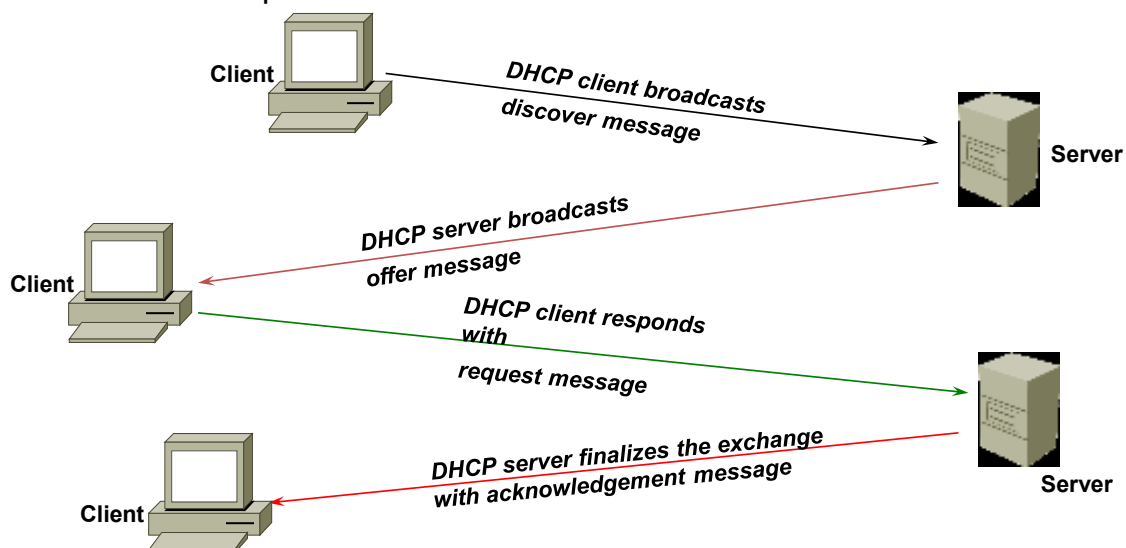


Connect your two routers together with a LAN connection, then copy the IOS with the **copy tftp flash** command.

Now this is a great feature of a Cisco router! If you do not have a laptop or other host that can provide TFTP services, you can make a router a TFTP server with the global configuration command “**tftp-server flash:**”. You will then need to make sure the file (i.e. IOS image) is available on the router you configured as a tftp server. It is as simple as that, you can now copy the image from the router configured as a tftp server allowing upgrade or downgrade of your other router.

Dynamic Host Configuration Protocol

DHCP – method to automatically assign IP Addresses along with other parameters



DHCP is used to assign IP addresses automatically and to set TCP/IP stack configuration parameters, such as the subnet mask, default router, and Domain Name System (DNS) servers. DHCP is also used to provide other configuration information as necessary, including the length of time the address has been allocated to the host. DHCP consists of two components: a protocol for delivering host-specific configuration parameters from a DHCP server to a host, and a mechanism for allocating network addresses to hosts.

Using DHCP, a host can obtain an IP address quickly and dynamically. All that is required is a defined range of IP addresses on a DHCP server. As hosts come online, they contact the DHCP server and request address information. The DHCP server selects an address and allocates it to that host. The address is only leased to the host, so the host will periodically contact the DHCP server to extend the lease. This lease mechanism ensures that hosts that have been moved or are switched off for extended periods of time do not hold on to addresses that they do not use. The addresses are returned to the address pool by the DHCP server, to be reallocated as necessary.

Making Your Router a DHCP Server

Issue the following commands to make your router a DHCP server

```
Router# config t
```

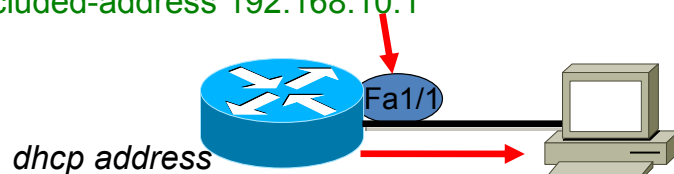
```
Router(config)# ip dhcp-pool LAN_A
```

```
Router(config-dhcp)# network 192.168.10.0 255.255.255.0
```

```
Router(config-dhcp)# default-router 192.168.10.1
```

```
Router(config-dhcp)# dns-server 63.10.1.1
```

```
Router(config)# ip dhcp excluded-address 192.168.10.1
```



This is another great Cisco router feature.

It is important that you understand that the router maps the pool to the interface which has an IP address in the same subnet as the pool.

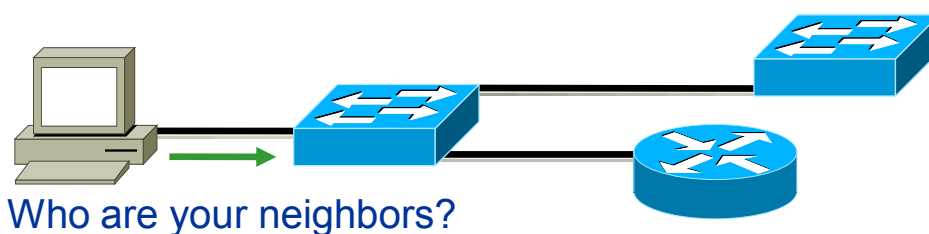
In the example above, the Fa1/1 interface must be assigned the IP address 192.168.10.1 or the pool will not hand out IP addresses to clients.

Note: The IP address of 192.168.10.1 is excluded from the pool since it is assigned to the router. Additional addresses can be excluded if they are specifically assigned to other devices.

Note: The network, default gateway and DNS server is defined as that information will be passed in the DHCP message to the host requesting to be assigned an address via DHCP.

Cisco Discovery Protocol

- Cisco Proprietary
- Gathers information about other Cisco neighbor devices only
- Turned on by default on all Cisco routers and switches
- Operates at layer two



Cisco Discovery Protocol (CDP) is a proprietary protocol designed by Cisco to help administrators collect information about both locally attached and remote devices. By using CDP, you can gather hardware and protocol information about neighbor devices, regardless of the routed protocols enabled on the interface since it operates at layer 2. This is very useful information for troubleshooting and documenting your Cisco-based networks. CDP is turned on by default on all Cisco routers and switches.

Cisco Discovery Protocol

Show commands

```
show cdp
show cdp neighbors
show cdp neighbors detail
show cdp entry *
show cdp interface
show cdp traffic
```

Global Config

```
cdp holdtime
cdp timer
cdp run
```

Interface

```
cdp enable
```

The “**show cdp neighbor**” command (**sh cdp nei** for short) delivers information about directly connected devices.

It’s important to remember that CDP packets aren’t passed through a Cisco switch, and that you only see what’s directly attached. So this means that if your a router is connected to a switch, you won’t see any of the devices hooked up to that switch, you will need to get that information from the switch itself.

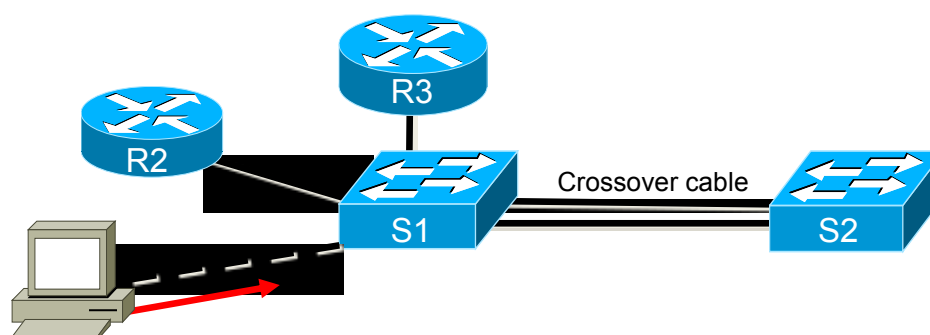
Another valuable CDP command to get more information about a neighbor is the “**show cdp neighbor detail**” command (**show cdp nei de** for short). This command can be run on both routers and switches, and it displays detailed information about each device connected to the device you’re running the command on.

The “**show cdp entry ***” command is the same as “**show cdp nei detail**”. However, on a router or switch, type “**show cdp entry * ?**” and you’ll see there are two helpful subcommands you can use.

show cdp neighbors

S1# **show cdp neighbors**

Device ID	Local Intrfce	Holdtme	Capability	Platform	Port ID
R2	Fas 0/1	170	R S I	Cisco 2811	Fas 0/0
R3	Fas 0/2	178	R	Cisco C804	Eth 0
S2	Fas 0/12	171	S I	WS-C3550-2	Fas 0/2
S2	Fas 0/11	171	S I	WS-C3550-2	Fas 0/1



Who are your neighbors?

Field Descriptions:

Device ID - The configured ID (name), MAC address, or serial number of the neighbor device.

Local Intrfce - (Local Interface) The protocol being used by the connectivity media.

Holdtime - (Holdtime) The remaining amount of time, in seconds, the current device will hold the CDP advertisement from a transmitting router before discarding it.

Capability - The capability code discovered on the device. This is the type of the device listed in the CDP Neighbors table.

Platform - The product number of the device.

Port ID - The protocol and port number of the device.

Command syntax:

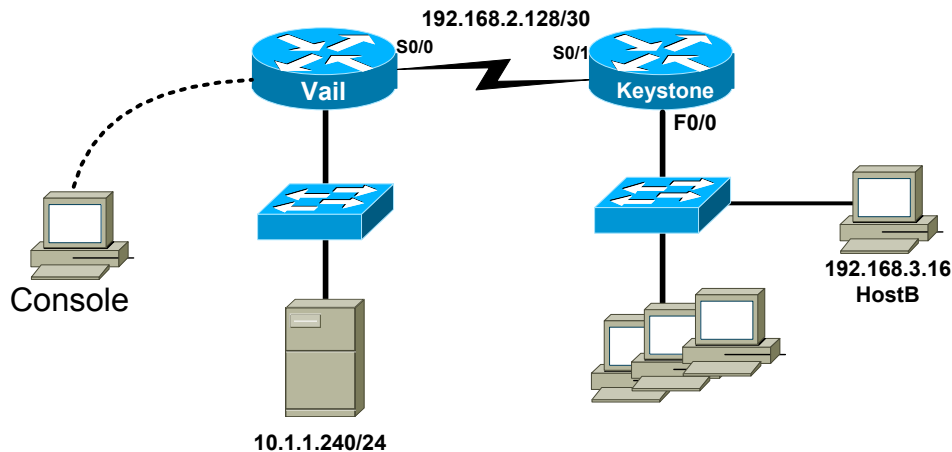
show cdp neighbors [*type number*] [**detail**]

type - (Optional) Type of the interface connected to the neighbors about which you want information.

number - (Optional) Number of the interface connected to the neighbors about which you want information.

detail - (Optional) Displays detailed information about a neighbor (or neighbors) including network address, enabled protocols, hold time, and software version.

Using CDP Example



You can only console into the Vail router and it is not configured....How can you get the Keystone's IP address so you can configure Vail with the correct IP address? In addition, HostB needs to be able to ping the server. How will this be accomplished?

1. You first need to administratively mark up the s0/0 interface on the Vail router so you can receive CDP information


```
Vail>enable
Vail#config t
Vail(config)#int s0/0
Vail(config-if)#no shutdown
```
2. You need to find the Keystone routers IP address and set the address of the Vail s0/0 to the next address in the available pool


```
Vail(config-if)#exit
Vail(config)#exit
Vail#show cdp neighbors detail
```
3. Once you find the IP address of the Keystone router, configure the Vail interface with the correct IP address – the next available IP address in the pool.
4. Telnet from the Vail router into the Keystone router and verify the configuration. Enable the F0/0 with a no shutdown if needed.
5. Finally, connect to HostB and make sure you can ping the server at 10.1.1.240.

Telnet

From a host prompt:
> telnet ip_address

From a router prompt:

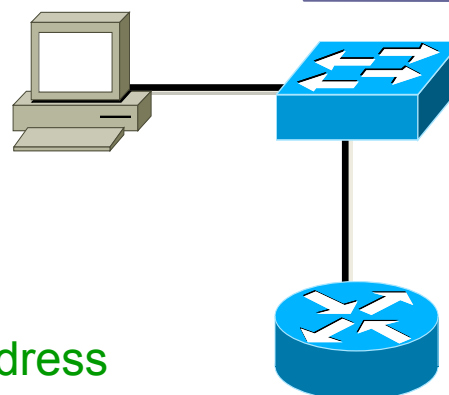
RouterA# telnet ip_address

Suspending and resuming a telnet session:

RouterB# [ctrl]-[shift]-6 then x

RouterA# show sessions

RouterA# resume <session#>



Telnet is a virtual terminal protocol that's part of the TCP/IP protocol suite that allows you to make connections to remote devices, gather information, and run programs. After your routers and switches are configured, you can use the Telnet program to reconfigure and/or troubleshoot your routers and switches without using a console cable.

You run the Telnet program by typing **telnet** from any command prompt (DOS or Cisco).

In order to be able to remotely telnet to your router or switch, you have to have the VTY passwords set. Otherwise, the router or switch will prompt that password is not set, and not permit the remote login.

If you telnet to a router or switch, you can end the connection by typing **exit** at any time, but what if you want to keep your connection to a remote device but still come back to your original router console?

To do that, you can press the Ctrl+Shift+6 key combination, release it, and then press X.

Another common practice is to telnet and specify a port or socket. This is useful when accessing a device hanging off of a terminal server, or when testing listener ports or firewall access rules.