# - Router Components -

## *Router Memory Components*

Cisco routers (and switches) generally contain four types of memory:
- **ROM (Read-Only Memory)**
- **Flash**
- **NVRAM (Non-Volatile RAM)**
- **RAM (Random-Access Memory)**

**ROM** contains a **bootstrap** program called **ROM Monitor (**or **ROMmon)**. When a router is powered on, the bootstrap runs a hardware diagnostic called **POST (Power-On Self Test)**.

If POST completes successfully, the bootstrap then attempts to locate and load the Cisco **IOS (Internetwork Operating System)** stored in **Flash memory.** Flash memory *can* be erased or overwritten, thus making the Cisco IOS upgradeable. The Cisco IOS is covered in great detail in other guides.

If the bootstrap *cannot* find the IOS in Flash, a stripped-down version of the IOS that will be loaded from ROM instead. The contents of ROM cannot be altered or erased; the entire ROM chip must be replaced if an upgrade/repair is necessary.

If the bootstrap *does* find the IOS in Flash, it is loaded into RAM and attempts to find a **Startup Configuration** (**startup-config)** file in **NVRAM**. NVRAM is non-volatile, thus its contents will survive a power-cycle.

If the IOS *cannot* find a startup-config file in NVRAM, it will attempt to load a configuration file from a **TFTP** server (this request is **broadcasted** to 255.255.255.255). If no TFTP server responds, the IOS will enter **Initial Configuration Mode**, a series of interactive questions intended for quick configuration of the router.

If the IOS *does* find a startup-config file in NVRAM, this file is loaded into **RAM**, and becomes the **Running Configuration (running-config)**. RAM is a volatile memory, and thus its contents will be lost if the router is power-cycled.

## *Router Memory, Quick Reference*

The following table details each of the basic types of **router memory**:

| *Memory* | *Writable?* | *Volatile?* | *Function* |
|---|---|---|---|
| ROM | No | No | *Stores bootstrap* |
| Flash | Yes | No | *Stores IOS* |
| NVRAM | Yes | No | *Stores startup-config* |
| RAM | Yes | Yes | *Stores running-config* |

## *The Router Boot-Process, Quick Reference*

The following details the **router boot process**:

1. The router is powered on.

2. The bootstrap program (ROMmon) is loaded from ROM.

3. The bootstrap runs POST.

4. The bootstrap attempts to load the IOS from Flash.
   a. If the IOS *is not* found in Flash, the bootstrap loads into RAM the basic IOS stored in ROM.
   b. If the IOS *is* found in Flash, it is loaded into RAM.

5. The IOS attempts to load the startup-config file from NVRAM
   a. If the startup-config *is not* found in NVRAM, the IOS attempts to load a configuration file from TFTP.
   b. If no TFTP server responds, the router enters Initial Configuration Mode.
   c. If the startup-config *is* found in NVRAM, it is loaded into RAM.

6. The startup-config becomes the running-config in RAM.

### *Interfaces vs Lines*

Cisco devices contain two distinctly different types of ports, **interfaces** and **lines**.

**Interfaces** connect routers and switches to each other. In other words, traffic is actually routed or switched across interfaces. Examples of interfaces include (but are not limited to):

- Serial interfaces
- Ethernet interfaces
- Fast Ethernet interfaces
- Token Ring interfaces
- ATM interfaces

Interfaces are identified by both the *type* of interface, and the **interface number** (which always begins at "0"). Thus, the first Ethernet interface on a router would be identified as *Ethernet0.*

Certain router families (such as the 3600 series) are modular, and have multiple "slots" for interfaces. Thus, interfaces on these routers are identified by both the **module number** *and* the interface number, formatted as: *module/interface.* Thus, the third Fast Ethernet interface on the first modular slot would be identified as *FastEthernet0/2.*

**Lines** identify ports that allow us to connect into, and then configure, Cisco devices. The most common examples of lines include:

- Console ports
- Auxiliary ports
- VTY (telnet) ports

Just like interfaces, lines are identified by both the *type* of line, and the line *number* (again, always begins at "0"). Thus, the first console port on a router would be identified as *Console0.*

### *Using Lines to Configure the IOS*

As mentioned previously, three methods (or *lines*) exist to configure Cisco IOS devices:
- Console ports
- Auxiliary ports
- VTY (telnet) ports

Nearly every modern Cisco router or switch includes a **console port,** sometimes labeled on the device simply as *con***.** The console port is generally a RJ-45 connector, and requires a **rollover** cable to connect to. The opposite side of the rollover cable connects to a PC's serial port using a serial **terminal adapter**.

From the PC, software such as HyperTerminal is required to make a connection from the local serial port to the router console port. The following settings are necessary for a successful connection:
- Bits per second - *9600 baud*
- Data bits - *8*
- Parity - *None*
- Stop bits - *1*
- Flow Control - *Hardware*

Some Cisco devices include an **auxiliary port,** in addition to the console port. The auxiliary port can function similarly to a console port, and can be accessed using a rollover cable. Additionally, auxiliary ports support modem commands, thus providing dial-in access to Cisco devices.

**Telnet**, and now **SSH**, are the most common methods of remote access to routers and switches. The standard edition of the IOS supports up to **5 simultaneous** VTY connections. Enterprise editions of the IOS support up to **255 VTY** connections.

There are two requirements before a router/switch will accept a VTY connection:
- An **IP address** must be configured on an interface
- At least one VTY port must be configured with a **password**