Real-Life Hardware: *Cisco*

- **Lab: Introduction to Cisco IOS**
  - Shiv Kalyanaraman
  - Kerry Wood (woodk@alum.rpi.edu)
  - Yong Xia (TA)
What is IOS?

- Internetwork Operating System
- A derivative of BSD UNIX
  - Custom built by Cisco for each platform
  - Pre-packaged and static.
  - Features available in different revisions (for a price!)
  - GUI’s available, but 90%+ of users still prefer command-line configuration.
iOS and Hardware

- iOS is designed to be hardware independent. A high end router may use ASIC’s for fast routing, while a smaller access router may use the central processor. Configuration and monitoring commands should be almost identical!

The OS not only configures the device, but is an abstraction to make it easier for humans!
Router Components RAM

RAM

- Temporary storage of config files
- All content is lost on power-down
- May be very large to hold large routing tables

Stores

- *Routing tables* (remember: usually built dynamically)
- *ARP cache* (again, built dynamically and ephemeral)
- *Fast-switching cache*
- *Packet buffers*
- *Packet hold queues*
**Router Components NVRAM**

**NVRAM**
- Non-volatile RAM
- Not especially fast
- Content is NOT LOST on power cycle.
- Usually less than 100 Kbytes

**Stores**
- *Exclusively used to store configuration scripts that are parsed on power-up.*
Router Components FLASH

FLASH
- EEPROM (Electronically Erasable Programmable Read-Only Memory)
- Retained on power-down.
- May be off-board in the form of flashcards.

Stores
- IOS versions. Allows for upgrading Operating System without replacing chip.
- Multiple versions of IOS may be stored at once (GREAT for testing install of a new version!)
- Off-board configuration allows you to “put OS in your pocket”.

Shivkumar Kalyanaraman
Router Components ROM

ROM
- Read-Only Memory.
- Generally installed in factory and never touched again!

Stores
- POST (power-on self test)
- Bootstrap program that calls IOS.
- Bare-bones version of an IOS. If IOS load routine fails, it defaults back to this version, giving limited functionality.
Boot Procedure

ROM init

TFTP?

FLASH

NVRAM

Contains location of IOS, could be FLASH, TFTP, etc.

CONFIG SCRIPT

IOS Version

RAM

Loaded IOS..

Config Params from script..

Route tables..

ARP..

Built during runtime

Shivkumar Kalyanaraman
IOS Interface

- IOS commands are very “assembly-like”. They are extremely granular, and often counter-intuitive.
- This is somewhat alleviated with inline help. The “?” character will become your best friend.
- Since these configurations are stored as text scripts, you can look up lots of examples on Cisco’s website.
In-line Help

- Typing the “?” character either after a command or at some point during a word brings up configuration parameters:

  For instance, typing:
  
  “ac?” <RETURN>

  Will ask the IOS what commands begin with “ac”, it will respond with:
  
  “access-enable access-template”

  This also works to find command modifiers or subsets:
  
  “show ip ?”
  
  Asks the IOS what is available for the “show ip” command:
  
  “interface route …”
Shortcuts

- Just like most *NIX’s, IOS has a wide array of shortcut keys to expedite configuration.

<table>
<thead>
<tr>
<th>Command</th>
<th>Issued At</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delete</td>
<td>Anywhere</td>
<td>Removes one char to the right of cursor</td>
</tr>
<tr>
<td>Backspace</td>
<td>Anywhere</td>
<td>Removes one char to the left of cursor</td>
</tr>
<tr>
<td>TAB</td>
<td>Anywhere</td>
<td>Finishes a partial command.</td>
</tr>
<tr>
<td>Ctrl-A</td>
<td>Anywhere</td>
<td>Moves the cursor to beginning of current line.</td>
</tr>
<tr>
<td>Ctrl-E</td>
<td>Anywhere</td>
<td>Moves cursor to end of current line.</td>
</tr>
<tr>
<td>Ctrl-R</td>
<td>Anywhere</td>
<td>Redisplays a line.</td>
</tr>
<tr>
<td>Ctrl-U</td>
<td>Anywhere</td>
<td>Erases a line.</td>
</tr>
<tr>
<td>Ctrl-W</td>
<td>Anywhere</td>
<td>Erases previous word.</td>
</tr>
<tr>
<td>Ctrl-Z</td>
<td>Configuration Mode</td>
<td>Ends configuration mode and returns to EXEC</td>
</tr>
<tr>
<td>Up Arrow</td>
<td>Anywhere</td>
<td>Scroll forward through former commands.</td>
</tr>
<tr>
<td>Down Arrow</td>
<td>Anywhere</td>
<td>Scroll backward through former commands.</td>
</tr>
</tbody>
</table>

- The most useful is the TAB key, which as you may know, completes a word.
Cisco 7000 Series

Cisco’s previous core layer router.

- 533-Mbps CxBus, 5 interface processor slots, 1 RP slot, and 1 SP (or SSP) slot


For more info.

This product is currently end-of-life, but is widely used.
Our 7000’s

We are currently in possession of three Cisco 7000’s donated to RPI by MCI WorldCom. All of the devices were pulled from WorldCom’s core network during their upgrade to Cisco’s 7500 line.

Advantages:
• Lots of varied hardware. Serial, fiber, Ethernet.
• Good, useful IOS versions.
• Solid, redundant configurations.

Disadvantages:
• Old and beat. When you turn these on, they sound like a K-car.
• Missing pieces. Ethernet AUI’s, serial connections.
A Single 7000 Configuration
Cisco 2500 Series

Cisco’s access-layer router.

- **Low performance, low cost.**
- **Memory switched architecture.**
- **Sadly, end of sale on this model. 2600 is a direct descendant/replacement.**

Check:

For more info.
Our 2500’s

In addition to the core devices, some access level devices were also donated. The early 2500 series were single configuration. We have 4 2503’s (hardwired configuration) and one configurable device (2524).

2503’s
- 1 AUI Ethernet Connector
- 2 50-pin Serial Connectors

2524
- 1 AUI / 10 Base-T Ethernet connector
- 3 Expansion slots (2 Serial connections and 1 ISDN BRI)
2500 Centerfold
Our Setup

Hidden deep within the recesses of the closet in the front of the room lies our Cisco equipment.

There are 5 2500 series Cisco routers above the 3 big 7000 series.

**Also visible in this picture are the Intel IXA boxes (but we’ll save those for later).
Connecting to The Routers

Because the routers have no configuration, we’ll connect to them via the console ports. Rather than sitting in the room with laptops, we’ll use a console server to set up a serial connection.

Terminal servers or modems are almost always attached to a router’s console port. That way you can access a malfunctioning device without being in the room.

Shivkumar Kalyanaraman

Rensselaer Polytechnic Institute
Why connect this way?

Until routers are configured, they generally have no connection to any medium. By default most interfaces are turned off. To perform the initial install, a serial connection is set up to the console port. Many times, this requires lugging your laptop into a data closet, and sitting on old dusty boxes while you work.

*Instead, we have a simple server that has a single IP address and 32 serial ports on the back. By telnet-ing to the IP of the box, and a specific port, you can access one of the devices.*
Telnet and Teams

Since this lab is a general introduction to the command-line, we can use all 8 routers. (Future labs are limited by connectivity issues). So, split up into 8 groups and then claim a router!

Available routers are:

7000 Series:
- 7000_1, telnet port: 2129
- 7000_2, telnet port: 2132
- 7000_3, telnet port: 2131

2500 Series:
- 2524_1, telnet port: 2127
- 2500_1, telnet port: 2123
- 2500_2, telnet port: 2124
- 2500_3, telnet port: 2125
- 2500_4, telnet port: 2126

To connect, telnet to:

`litec-wti.ecse.rpi.edu <PORT_NUM>`
Lab Goals

- What will we learn in this lab?
  - A brief overview of the Cisco design philosophy.
  - How one can access a Cisco router.
  - The Cisco security model.
  - Command line usage and hints.
  - How to build a configuration script and store it in NVRAM.
  - What the CLI looks like and how it interacts.
  - What it feels like to get your hands on some real hardware!
Lab References

- Useful References:
  - http://www.cisco.com/