T-110.5101 Laboratory works

Assignment 6: Router
Cisco Internetworking Operating System
Fall 2012

Jere Mäkelä
8.7.2011
updated 10.7.2012 Sipi Seppälä

Table of Contents

General

- Goal:
  - To be able to do the basic L2/L3 configurations with Cisco IOS devices (switch, router)
  - To familiarize with Cisco CLI interface (Command Line Interface)
  - The commands invoked, for the most part, belong to CCNA requirements
  - If you want to become proficient with Cisco IOS, the best place to start with is one of the reference books and buy a simulator, if you don't have an access to Cisco devices
- Preassignment
  - Subnetting with VLSM. The preassignment is in an Excel file ”Preliminary Task”. Preassignment is worth 2 points
- User commands:
  - The user typings are in **bold text**

CCNA requirements

- CCNA knowledge base is a solid foundation, which gives you a reasonable knowledge to work with small and medium-size networks
- Passing CCNA certificate is quite a big task. You need to memorize a 1000p book and you need to be able to subnet in your head and fast!
- CCNA topics:
  - Internetworking, OSI model, TCP/IP model
  - Subnetting, VLSM (Variable length subnet mask) and TCP/IP troubleshooting
  - Cisco IOS and SDM (Security Device Manager: browser based user interface)
  - Managing a Cisco network
  - IP routing (static routing, RIP, IGRP, EIGRP, OSPF)
  - Spanning Tree Protocol, EtherChannel (L2 redundancy and loop-free topology)
  - Virtual LANs (VLAN)
  - Security (ACL – Access Control List)
  - NAT (Network Address Translation)
  - Cisco Wireless technologies
  - IPv6
  - WAN (Wide Area Networks)

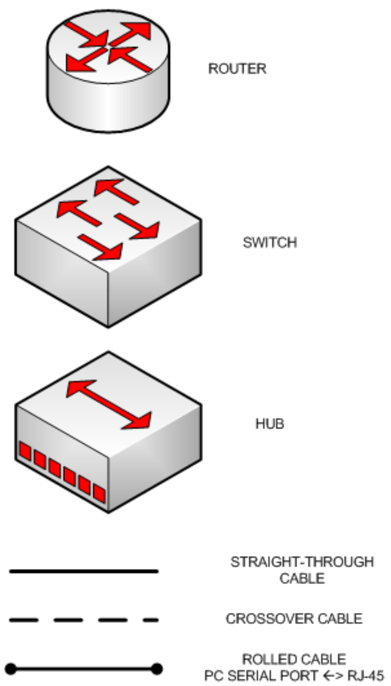
Laboratory

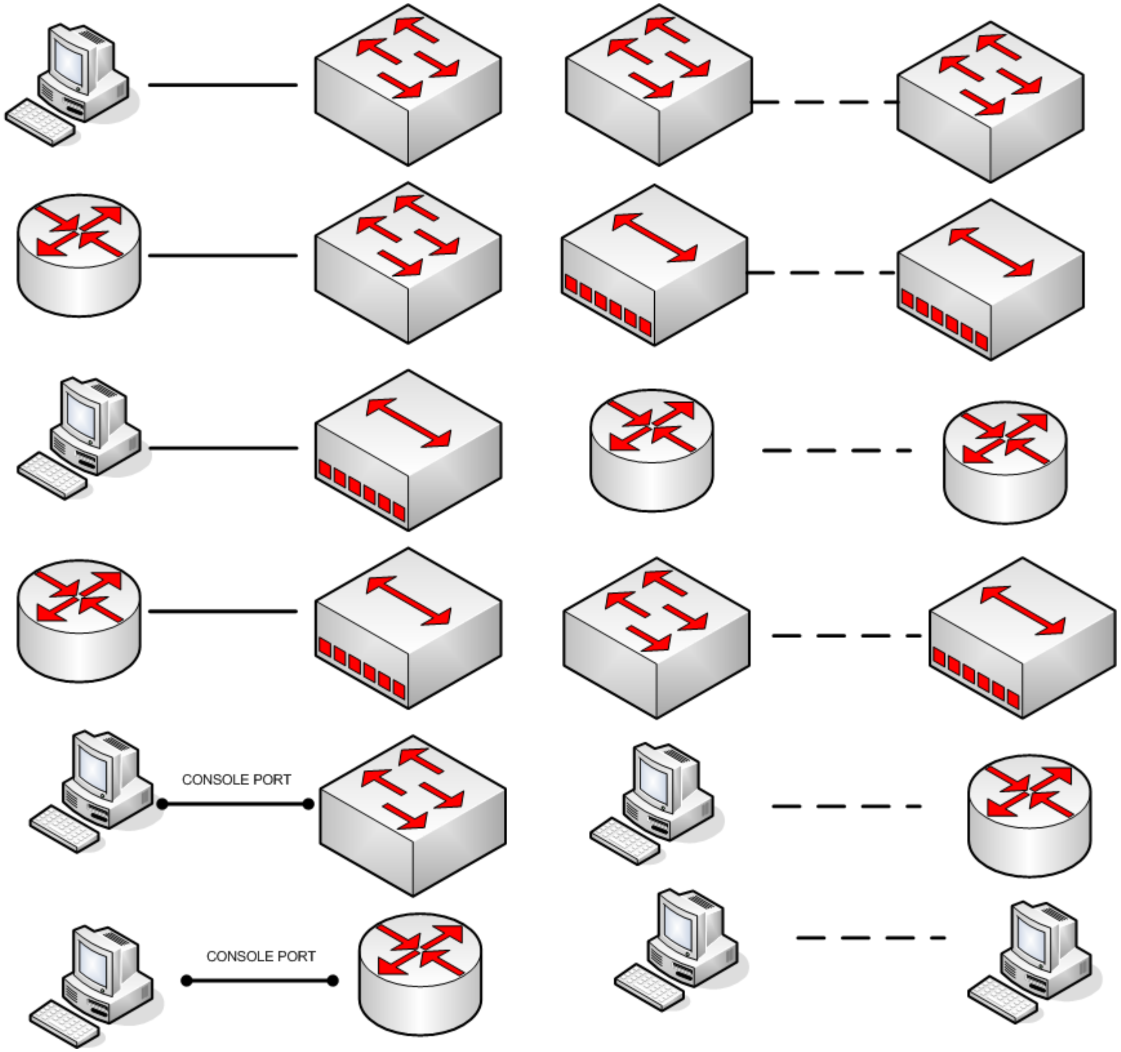
- The IOS commands are done with a simulator:
  - CCNA Network Visualizer 6.0
  - The commands supported by the simulator is a subset of CCNA commands. Some commands do not work correctly or not at all
  - If you have to check the commands supported, consult:
    - http://www.routersim.com/CCNA6_Supported_Commands.html
- The switch model 2960 and the router model 2811 are employed in tasks
- In order to do something useful with Cisco devices, you need to know a whole lot of things before delving into configurations. This makes this assignment challenging and tedious
- If you want to do the easy way, you can copy&paste from the lab book ”CCNA Portable Command Guide” but what is the use?

Laboratory works

- The tasks should be considered more as academic examples rather than feasible real-world implementations of a network
- #1 Basic Switch and Router configurations, DHCP and port security
  - The reason we configure telnet instead of SSH is that the simulator only supports telnet sessions
  - 6p
- #2 Routing
  - Static routing and RIPv2 are employed. Other routing protocols, such as OSPF, are harder to configure and were left outside because of a short time per lab
  - 8p
- #3 NAT and ACL
  - 8p
- The maximum points from this lab is 22p
- If you are well prepared, each work should take no more than 30-45 minutes making this a 2-3 hour assignment
- Write down the complete IOS command sequence for each task before coming to the lab. Be particularly accurate, in which Cisco IOS mode you have to be and how to move between the modes. Before invoking the commands, you should check first the CCNA Portable Command Guide at the lab
- Outside the lab times, a group can borrow the CCNA Portable Command Guide for three hours
- The reference book: Richard Deal: CCNA Study Guide. 3rd edition can also be borrowed if some things in these slides stay blur

Cisco cabling

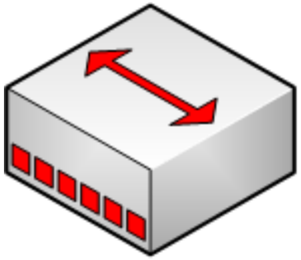




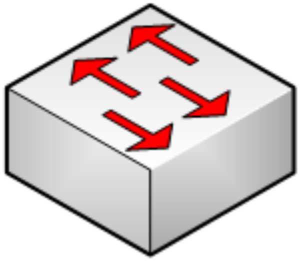
Connectivity problems

- You do need a crossover cable between Cisco switches. This is a common source of connectivity failures. If CDP (Cisco Discovery Protocol) shows no a neighbor present, even the power is on in both devices and there is a cable connecting these two, check that the cable is a crossover cable

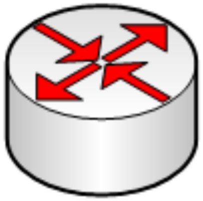
Collision and broadcast domains



HUB MAKES UP ONE SINGLE COLLISION DOMAIN



SWITCH PORTS MAKE UP ONE SINGLE COLLISION DOMAIN. I.E. A SWITCH WITH 8 PORTS HAS 8 COLLISION DOMAINS



ROUTER INTERFACE IS ONE SINGLE BROADCASTING DOMAIN.
CISCO SAYS, THAT THE MAXIMUM BROADCASTING SPACE IS 500 HOSTS BEFORE RUNNING INTO BROADCASTING PROBLEMS

A ROUTER DOES NOT PROPAGATE LOCAL BROADCASTS

Cisco configuration

- Connections:
  - Console port, Auxiliary Port, Telnet, SSH, Browser (SDM), SNMP, Cisco Works and Cisco Managed Services Solutions
  - Telnet and SSH are called VTY access (virtual type terminal)
- Console port:
  - Hyper Terminal or PuTTY
  - Speed: 9600 bps
  - Data bits: 8
  - Stop bits: 1
  - Parity: None
  - Flow Control: None

Cisco interface nomenclature (1/3)

- Switch 2950, 2960
  - Fixed interfaces
  - ”Type slot#/port#”
    - Type: ethernet (10M), fastethernet (100M), gigabit
    - Slot# is always 0
    - Port# starts from 1
    - i.e. ”fast 0/1” or ”f0/1”
- Note, that you can shorten just about any IOS command or option provided that the abbreviated command is non-ambiguous

Cisco interface nomenclature (2/3)

- Routers and some switches (6500)
  - Fixed or modular interfaces
  - For fixed interfaces:
    - ”Type port#”
      - Type: atm, asynch, bri, ethernet, fastethernet, gigabitethernet, serial
      - Port# starts from 0
      - i.e. ”serial 0” or ”s0”
  - For modular interfaces:
    - ”Type slot#/port#
      - Type: atm, asynch, bri, ethernet, fastethernet, gigabitethernet, serial
      - Slot# starts from 0
      - Port# starts from 0
      - i.e. ”giga 0/0”, ”fast 0/1” or ”f0/1”

Cisco interface nomenclature (3/3)

- To enter into an interface configuration mode, you have to type one of the following:
  - Router(config)#**interface ethernet 0/1**
  - Router(config)#**interface ethernet0/1**
  - Router(config)#**int e 0/1**
  - Router(config)#**int e0/1**
  - Note, that you must be in the global configuration mode before entering an interface configuration
- Router interfaces are shut down by default, switch ports are open by default. The VLAN interfaces of a switch are also shut down by default
  - To open the interface, type:
    - Router(config-if)#**no shutdown**
  - To shut down the interface, type:
    - Router(config-if)#**shutdown**
- You can view all the physical interfaces of a router or a switch by typing:
  - Router#**sh ip interface**
- Cisco IOS commands are negated or revoked by typing ”no” before the command

Cisco IOS modes (1/5)

- Login
  - Upon connecting to the console port
  - Hit **Enter** to enter the User Exec
- User Exec
  - The prompt becomes: Router>
  - Basic and limited access to IOS
  - Simple monitoring and troubleshooting such as: ”**show ?**”, ”**telnet**”, ”**ping**”, ”**traceroute**”
  - Type **”**Router>**enable”** (or ”Router**>en**” for short) to enter Privileged Exec
  - Type ”**logout**” or ”**exit**” to return to login
- Privileged Exec
  - The prompt becomes: Router#
  - High-level management to IOS
  - Includes all the commands from User Exec
  - You can do most things except configuring the device
  - Type **”**Router#**configure terminal**” (or ”Router#**conf t**” for short) to enter the configuration mode
  - Type ”**disable**” or ”**disa**” to return to User Exec
  - Type ”**logout**” or ”**exit**” to return to login
  - **Reload** to boot up the device

Cisco IOS modes (2/5)

- Configuration or global configuration mode
  - The prompt becomes: Router(config)#
  - Debug
  - Hostname
  - Enable secret
  - Ip route
  - ACL
  - Type **exit**, end or **cntl-z** to return to Privilege Exec

Cisco IOS modes (3/5)

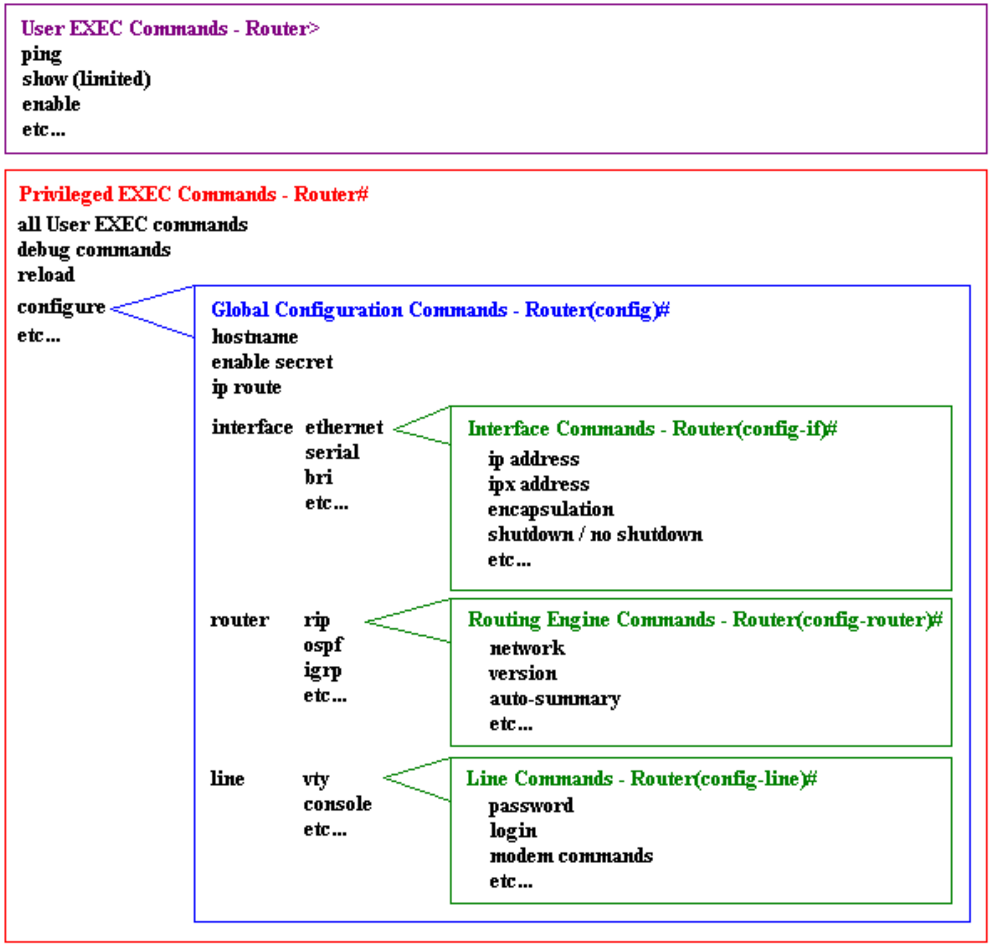
- To enter the interface configuration mode:
- Router(config)#**interface fastethernet 0/1** - OR - Router(config)#**int f0/1**
  - The prompt becomes: Router (config-if)#
    - Ip address + mask
    - Encapsulation
    - shutdown / no shutdown (or shut for short)
    - Type exit to return to Configuration mode, end or cntl-z to return to Privileged Exec mode
- Router engine commands
- Router(config)#**router rip|ospf|igrp|eigrp**
  - The prompt becomes: Router(config-router)#
  - Network etc.
- Line commands i.e.
- Router(config)#**line con 0**
  - The prompt becomes: Router(config-line)#
  - Password
  - Login
- Line console is the CLI interface to the device. The serial port configuration was introduced earlier. The cable type is: ”Serial (PC) – RJ45”

Cisco IOS modes (4/5)

- Telnet settings:
  - Router(config)#**line vty 0 ?**
  - The prompt becomes: Router(config-line)#
  - Type the question mark, if you don't know the number of VTY lines. The number of VTY lines is the number of maximum allowed, simultaneous Telnet sessions
    - Password
    - Login
- SSH configuration is not included in this course. But in the real life, SSH should be preferred over Telnet because of the security
- SSH configuration is a very straightforward operation

- You can always exit from any mode by typing ”**exit**”
- **cntl-z** returns straight back to the privileged exec mode

Cisco IOS modes (5/5)



User EXEC Commands - Router>
ping
show (limited)
enable
etc...

Privileged EXEC Commands - Router#
all User EXEC commands
debug commands
reload
configure
etc...

Global Configuration Commands - Router(config)#
hostname
enable secret
ip route

interface ethernet
serial
bri
etc...

Interface Commands - Router(config-if)#
ip address
ipx address
encapsulation
shutdown / no shutdown
etc...

router rip
ospf
igrp
etc...

Routing Engine Commands - Router(config-router)#
network
version
auto-summary
etc...

line vty
console
etc...

Line Commands - Router(config-line)#
password
login
modem commands
etc...

Figure: http://www.cisco.com/warp/cpropub/45/tutorial.htm

Cisco commands (1/2)

- Context-sensitive help:
  - Router>**?**
    - Shows all the possible commands in this mode
    - Type Spacebar to scroll down one page at a time
    - Type Enter to scroll down one line at a time
  - Router>**e?**
    - Shows all the commands starting with ”e”
  - ^ invalid input detected
  - Hit UP and DOWN ARROW to browse command history
  - To negate the command, you just typed, browse the command from the command history, type **ctrl-a** (to place the cursor in the beginning of the line) and type ”**no**” before the command
  - Router#**clock ?**
    - set Set the time and date
  - Router#**clock set ?**
    - Hh:mm:ss Current Time
    - Router#**clock set 15:00:00 ?**
    - <1-31> Day of the month
    - MONTH Month of the year
    - Router#**clock set 15:00:00 17 Mar ?**
    - <1993-2035> Year
    - Router#**clock set 15:00:00 17 Mar 2011**
    - Router#**show clock**

Cisco commands (2/2)

- The most important hot keys:
  - **cntl-a – Moves the cursor to the beginning of the line**
  - **cntl-e – Moves the cursor to the end of the line**
  - **up arrow – Recalls the last command**
  - **down arrow – Recalls the most previously executed command**
  - **tab – IOS completes the word (if the characters typed form a unique start for a command)**
  - **? – Presents all the possible commands or parameters**

- The rest of the hot keys:
  - esc-b – Moves the cursor back one word at a time
  - esc-f – Moves the cursor forward one word at a time
  - left arrow – Moves the cursor back one character at a time
  - right arrow – Moves the cursor forward one character at a time
  - cntl-d – Deletes the character the cursor is under
  - backspace – Deletes the character preceding the cursor
  - cntl-r – Redisplays the current line
  - cntl-u – Erases the line completely
  - cntl-w – Erases the word the cursor is under
  - cntl-z – Returns to the Privilege Exec mode from Configuration mode
  - $ - Indicates that there are more characters to the right of the $

Cisco configuration storage (1/4)

- running-config
  - RAM
  - Working configuration
  - Stores all the configurations you invoke
- startup-config
  - NVRAM or flash
  - non-volatile configuration
  - After reload or power off, the device copies ”startup-config” into ”running-config”
- Typically you want to save the given configurations into the non-volatile memory by:
  - Router#**copy run start**

Cisco configuration storage (2/4)

- Configuration register
  - In NVRAM
  - Register affects how the router boots up
  - ”Router>**sh version**” shows the IOS version, system image file and the configuration register value
  - ”Router(config)#**config-register 0xHEX_VALUE**” alters the register content
  - The default value is 0x2102
  - With the configuration register you can recover the password, in case you have forgotten it
- Flash
  - Default location of IOS images
  - Can have backup config files
  - Switch#**sh flash**

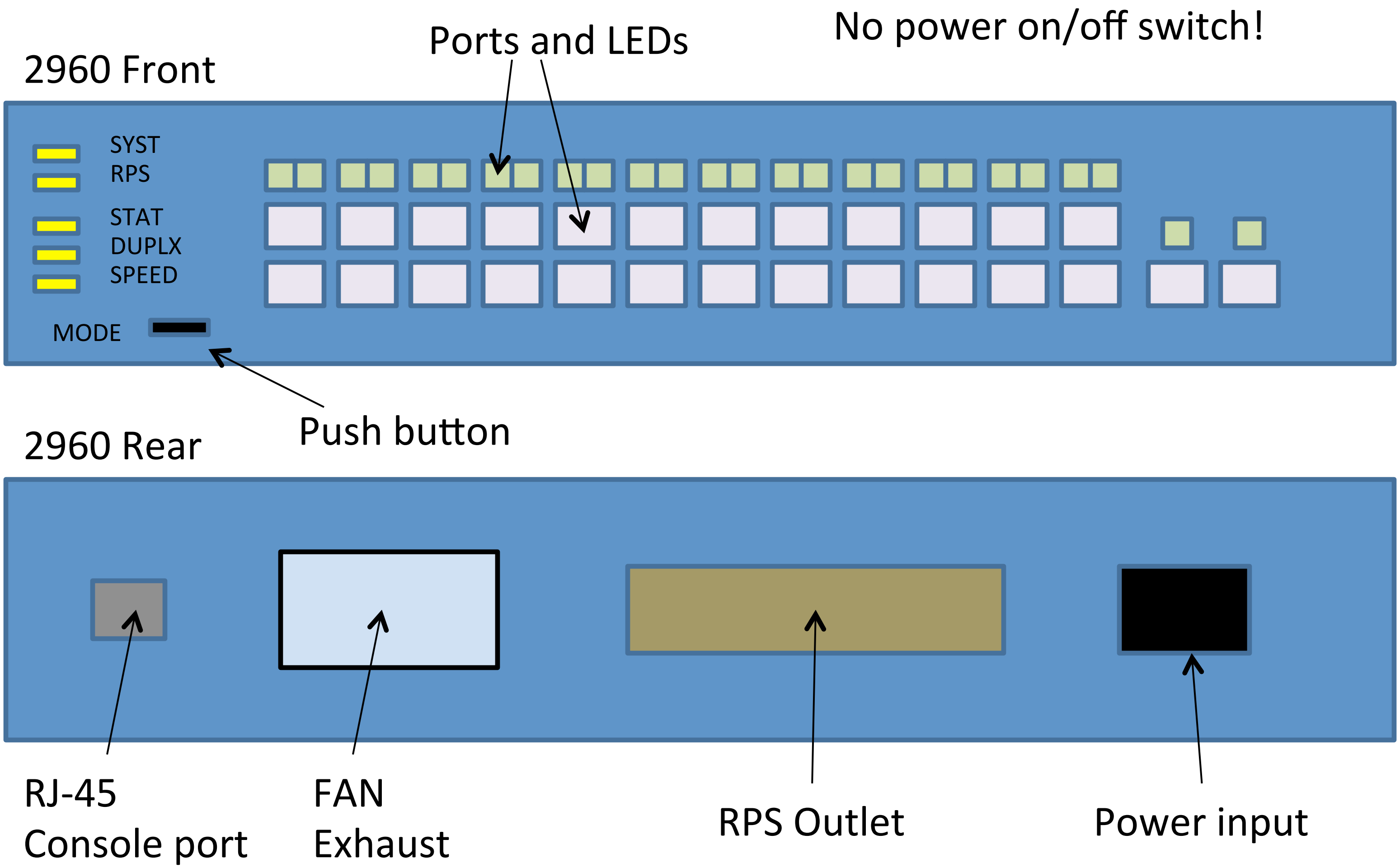
Cisco configuration storage (3/4)

- Router#**show run**
- Router#**show start**
- Router#**copy run start** (from run to start)
- Router#**copy start run** (from start to run. Obs! A merge operation, not a replacement)
- Router#**copy start tftp** (copy start to TFTP server)
- Router#**copy tftp run** (copy run from TFTP server)
- Router can have multiple copies of configuration files with different names in the flash memory. However, it is more advisable to copy them to a tftp/ftp server
  - Router#**copy run|start flash:FILE_NAME**

Cisco configuration storage (4/4)

- To wipe out the router configuration:
  - Router#**erase start**
  - Router#**reload**
- To wipe out the switch configuration:
  - Switch#**erase start**
  - Switch#**delete vlan.dat**
  - Switch#**reload**
- To wipe out i.e. a backed-up config:
  - Switch#**delete flash:FILE_NAME**
- To back up an IOS image:
  - Switch#**copy flash tftp**
- To load a new IOS image:
  - Switch#**copy tftp flash**
  - Switch#**reload**

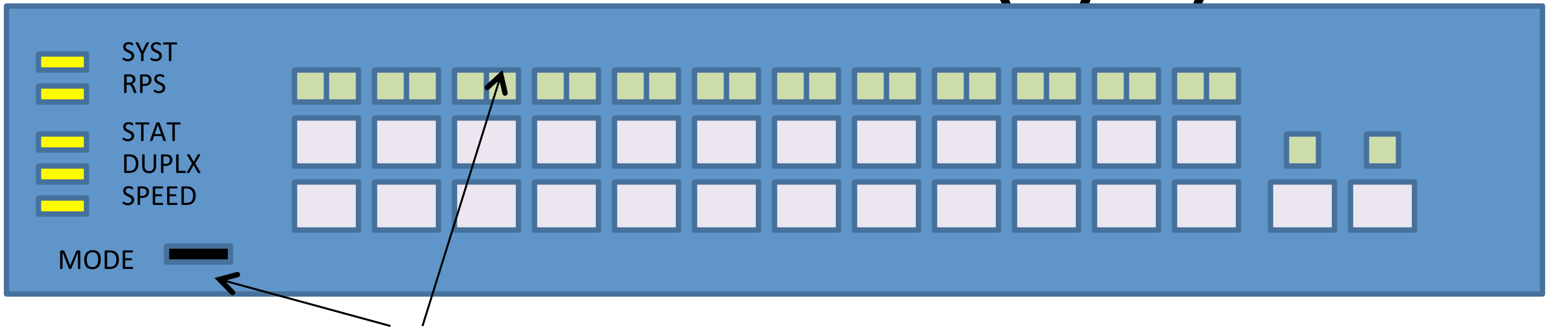
Cisco devices (1/3)



Cisco devices (2/3)

- System LED
  - Green: The system is up and running
  - Amber: The system has experienced a malfunction
  - Off: The system is powered down
- RPS (Redundant power system)
  - Green: RPS is attached and operational
  - Amber: RPS is installed but not operational
  - Flashing amber: Both the internal and external (RPS) installed, but RPS is providing power
  - Off: RPS is not installed

Cisco devices (3/3)



The meaning of the LED above the port depends on the LED's Mode setting.

When the Stat LED is lit, the port LEDs show the port status
- Green: A powered-up host connection
- Flashing green: Traffic is running in the port
- Flashing green and amber: An operational problem
- Amber: The port has been manually disabled, is in a blocking STP state or disabled because of a security breach

If you push the MODE button once, the MODE LED will change to Dublx. The port LEDs reflect the dublex setting:
-Off: Half-duplex
-Green: full-duplex

By pressing the MODE button again, the MODE LED will change to Speed. The port LEDs reflect the speed setting:
-Off: 10 Mbps
- Green: 100 Mbps
- Blinking green: 1000 Mbps

Cisco device bootup

- After power up:
  - #1 Flash is validated
  - #2 IOS is found, uncompressed and loaded. Note that there can be more than one IOS image. By default, the first IOS image is loaded but can be changed with ”boot system flash” command
  - #3 POST checks different components to see that they are operational (takes about a minute). First, system LED is off. The system LED turns into green if everything is ok. The amber usually is catastrophic to a switch
  - #4 Configuration is found and applied
  - #5 User is presented the User Exec prompt if hooked with a terminal emulator
  - If a configuration is missing in NVRAM, the switch starts up a setup script. You can start a setup later with a ”setup” command in Privilege Exec. With setup, however, you can do only the very basics
  - You can reload IOS by ”reload” command (remember to save your config before that:
    - Switch#**copy run start**
    - Switch#**reload**

Port security (1/2)

- Starting in IOS 12.1 (the latest commercial IOS version is 15.0)
- If and when you want to harden Cisco IOS, the port security is in your toolbox
- Works only with access ports, not with:
  - Trunk port, Switch port analyzer port or EtherChannel port
- Access port connects a host (or several hosts, if a hub is present). Trunk port connects between switches or routers
- Setup per interface as follows:

Port security (2/2)

- Configure a switch port to an access port with a VLAN number
  - Switch(config)#**int fast 0/0**
  - Switch(config-if)#**switchport mode access**
  - Switch(config-if)#**switchport access vlan VLAN#**
- Enable port-security to this port
  - Switch(config-if)#**switchport port-security**
- Define the maximum number of hosts, that can be connected to this switch port (typically 1)
  - Switch(config-if)#**switchport port-security maximum VALUE**
- Define, what will be the action, if the port-security is violated (too many hosts or a host with non-allowed MAC-address is found). Typically you want to shut down the port
  - Switch(config-if)#**switchport port-security violation protect|restrict|shutdown**
- After a shutdown:
  - Switch(config-if)#**no shutdown**
- Attach a MAC-address of a legal host
  - Switch(config-if)#**switchport port-security mac-address MAC-ADDRESS**
- OR
- Define the mac-address to be sticky, which attaches the first encountered mac-address
  - Switch(config-if)#**switchport port-security mac-address sticky**
- Check the port-security settings:
  - Switch#**sh port-security**

Basic switch configuration (1/6)

- Configure:
  - Switch name
  - Clock
  - Secret password (encrypts the password in the configuration file). Required, when entered into the privileged mode
  - Login banner. The text that a user gets, when logging in the device. Do NOT use a word ”welcome”. This is an invitation to a hacker and might lead in troubles if you have to go to the court
  - Disable DNS queries. DNS query is done and telnet invoked, if the IOS does not recognize the command you type. This is pretty annoying when you do a typo. The command is: ”no ip domain-lookup”
  - Console port password + login
  - ”logging synchronous” command, which inhibits the garbled text on a command line
  - The idle time, after which the console logs off
  - Note, that the simulator does not support ”**no ip domain-lookup**” or ”**logging synchronous**” commands. You have to know these nevertheless
  - VTY password + login. The number of allowed VTY connections depends on the switch model

Basic switch configuration (2/6)

- Configure (cont.):
  - IP default gateway. This is the IP address of the router's interface that connects into this switch
  - Management VLAN. All the switches are configured to belong into a single management VLAN. That is one common subnet. The IP address of the management VLAN is required, when the switch is configured with VTY. Note that the VLAN interface is shut by default, so you need to open (enable) it with "**no shut**" command
  - By default, all the switch ports belong to VLAN 1. If you don't configure ports, they belong by default to VLAN 1 and are access ports
  - Port-security settings. You must configure the switch port to an access port (vs. trunk). Then you enable the port-security. Then you can set the maximum number of hosts, that can be connected into this port (number of MAC addresses). You can also set the switch port to be sticky, that is, you don't have to program the allowed MAC address (or addresses) but let the switch learn the first encountered MAC address on a port and attach it to the allowed address. You also configure the action that is done after the port violation. Typically this would be shutting down the port
  - Finally, copy running-config into startup-config with "Router#**copy run start**"

Basic switch configuration (3/6)

- Switch Con0 is now available
- Press RETURN to get started!
- Switch>**en**
- Switch#**conf t**
- Enter configuration commands, one per line. End with CNTL/Z
- Switch(config)#**no ip domain-lookup**
- ^
- % Invalid input detected at '^' marker. OBS! Does not work in the simulator
- Switch(config)#**hostname MY_SWITCH**
- MY_SWITCH(config)#**enable secret cisco**
- MY_SWITCH(config)#**banner login $**
- Enter TEXT message. End with the character '$'.
- **Authorized personnel only!**
- **Violators will be prosecuted**
- **to the fullest extent of the law.**
- **$**

Basic switch configuration (4/6)

```
MY_SWITCH(config)#line
MY_SWITCH(config)#line ?
  <0-16>   First Line number
  console  Primary terminal line
  vty      Virtual terminal

MY_SWITCH(config)#line con
MY_SWITCH(config)#line console 0
MY_SWITCH(config-line)#logging synchronous
             ^
% Invalid input detected at '^' marker. OBS! Does not work in the simulator
MY_SWITCH(config-line)#passwor
MY_SWITCH(config-line)#password cisco
MY_SWITCH(config-line)#login
MY_SWITCH(config-line)#exec-timeout 5 0
             ^
% Invalid input detected at '^' marker. OBS! ´Does not work in the simulator
MY_SWITCH(config-line)#exit
MY_SWITCH(config)#line
MY_SWITCH(config)#line ?
  <0-16>   First Line number
  console  Primary terminal line
  vty      Virtual terminal

MY_SWITCH(config)#line vty 0 ?
  <1-15>  Last Line number
  <cr>

MY_SWITCH(config)#line vty 0 15
```

Basic switch configuration (5/6)

```
MY_SWITCH(config-line)#password cisco
MY_SWITCH(config-line)#login
MY_SWITCH(config-line)#exit
MY_SWITCH(config)#ip de
MY_SWITCH(config)#ip default-gateway 10.0.0.1
MY_SWITCH(config)#inte
MY_SWITCH(config)#interface vlan 10
MY_SWITCH(config-if)#ip add
MY_SWITCH(config-if)#ip address 10.0.0.2 255.255.255.0
MY_SWITCH(config-if)#no shut
MY_SWITCH(config-if)#exit
MY_SWITCH(config)#int
MY_SWITCH(config)#interface fas
MY_SWITCH(config)#interface fastethernet 0/1
MY_SWITCH(config-if)#des
MY_SWITCH(config-if)#description Link to Moscow
MY_SWITCH(config-if)#swi
MY_SWITCH(config-if)#switchport por
MY_SWITCH(config-if)#switchport port-security
Command rejected: Not eligible for secure port.
MY_SWITCH(config-if)#swit
MY_SWITCH(config-if)#switchport mode
MY_SWITCH(config-if)#switchport mode acc
MY_SWITCH(config-if)#switchport mode access
MY_SWITCH(config-if)#switchport access vlan 10
MY_SWITCH(config-if)#switchport port-security
```

VLAN interface of a switch is shut down by default. Open it with ”no shut” commannd

Basic switch configuration (6/6)

- MY_SWITCH(config-if)#**swi**
- MY_SWITCH(config-if)#**switchport por**
- MY_SWITCH(config-if)#**switchport port-security max**
- MY_SWITCH(config-if)#**switchport port-security maximum 1**
- MY_SWITCH(config-if)#**swi**
- MY_SWITCH(config-if)#**switchport por**
- MY_SWITCH(config-if)#**switchport port-security viol**
- MY_SWITCH(config-if)#**switchport port-security violation shu**
- MY_SWITCH(config-if)#**switchport port-security violation shutdown**
- MY_SWITCH(config-if)#**swi**
- MY_SWITCH(config-if)#**switchport por**
- MY_SWITCH(config-if)#**switchport port-security mac**
- MY_SWITCH(config-if)#**switchport port-security mac-address stic**
- MY_SWITCH(config-if)#**switchport port-security mac-address sticky**
- MY_SWITCH(config-if)#**cntl-z**
- MY_SWITCH#c**opy run start**
- Destination filename [startup-config]?
- Building configuration...
- [OK]
- MY_SWITCH#

Running-config (1/3)

```
•   MY_SWITCH Con0 is now available
•   Press RETURN to get started!
•   Authorized personnel only!
•   Violators will be prosecuted
•   to the fullest extent of the law.
•   User Access Verification
•   Password:  password
•   MY_SWITCH>en
•   Enter password: *****
•   MY_SWITCH#sh run

•   Building configuration...

•   Current configuration : 918 bytes
•   !
•   version 12.2
•   no service pad
•   service timestamps debug uptime
•   service timestamps log uptime
•   no service password-encryption
•   !
•   hostname MY_SWITCH
•   !
•   enable secret 5 $1$u76B$IOFVJ7VxfVXYVpGDrFTcI0
```

Running-config (2/3)

```
no aaa new-model
system mtu routing 1500
no ip subnet-zero!
no file verify auto
spanning-tree mode pvst
spanning-tree extend system-id
!
vlan internal allocation policy ascending
!
interface FastEthernet0/1
 description "Link to Moscow"
 switchport mode access
 switchport port-security
 switchport port-security maximum 1
 switchport port-security mac-address sticky
!
interface FastEthernet0/2
interface FastEthernet0/3
interface FastEthernet0/4
interface FastEthernet0/5
interface FastEthernet0/6
interface FastEthernet0/7
interface FastEthernet0/8
interface GigabitEthernet0/1
```

Running-config (3/3)

```
interface Vlan10
  ip address 10.0.0.2 255.255.255.0
  no ip route-cache!
ip default-gateway 10.0.0.1
ip http server!
control-plane!
banner login ^C
Authorized personnel only!
Violators will be prosecuted
to the fullest extent of the law.
 ^C
line con 0
  password cisco
  login
line vty 0 4
  password cisco
  login
line vty 5 15
  password cisco
  login
end
MY_SWITCH#
```

Basic Switch configuration (cont.)

- In order to access other devices in other VLANs, or allow VTY connection to this host, an IP address and a default gateway must be configured with a switch
- In switches, the IP address settings are done per VLAN (virtual lan), not physical interfaces
- VLAN 1 is the default management VLAN. The management protocols of a switch (CDP, VTP, DTP) occur within the switch's management VLAN
- You should use a different VLAN# than 1 to manage your switches. Always use the same management VLAN# in all switches, for example VLAN 10
- For clarity, name your management VLAN as ”Management”
- In routers, the IP address settings are done for interfaces

Basic router configuration

- Very much like that of the switch, except that you configure each interface with IP settings
- You don't configure "ip default-gateway" with routers and you don't configure management VLAN
- Each interface of a router is a separate network of a subnet. The IP address of the router interface becomes the default gateway for the subnet. Typically, you choose the first available IP address for a router interface
  - Router#**conf t**
  - Router(conf)#**int f0/0**
  - Router(conf-if)#**ip address 192.168.1.1 255.255.255.0**
  - Router(conf-if)#**no shut**
  - Router(conf-if)#**exit**
  - Router(conf)#**int serial0**
  - Router(conf-if)#**ip address 192.168.2.1 255.255.255.0**
  - Router(conf-if)#**no shut**
  - Router(conf-if)#**clock rate 64000**
  - Router(conf-if)#**cntl-z**
  - Router#**copy run start**
- Note, that "clock rate" command is applied only with serial interfaces that have a DCE cable plugged into it. DTE (Data termination equipment) and DCE (Data communications equipment) are typically used in WAN connections
- Use "**show interfaces**" or "**show ip interfaces**" to verify the configuration
- EIGRP and OSPF need a bandwidth value for a router interface:
  - Router(config)#**int serial 0/0**
  - Router(config-if)#**bandwidth rate_in_kbps**
- Name resolution is done either statically (name to IP) or dynamically (DNS)
  - **ip name-server IP_address_of_DNS_server**
- Typically, DNS is disabled in routers:
  - Router(config)#**no ip domain-lookup**
- Router#**show hosts**

Basic troubleshooting (1/4)

OSI Reference Model Layer	**Command**
L2	show ip arp
L2	show interfaces
L2	show cdp neighbors
L3	ping
L3	traceroute
L7	telnet
L2-L7	debug

Basic troubleshooting (2/4)

- Switch#**sh ip arp**
  - Output shows the IP – MAC bindings and the interface
- CDP – Cisco Discovery Protocol
  - CDP shows information only from the directly connected devices
  - You can test L2 connectivity with CDP
  - The following info is gathered:
    - Name of the device (hostname)
    - IOS version
    - HW capabilities (routing, switching, bridging)
    - HW platform, such as 2960
    - L3 addresses of the device
    - The interface on which the CDP update was generated
  - Switch#**sh cdp neighbors**
  - Switch#**sh cdp nei detail**
  - You can disable or enable CDP globally or per port. You should disable CDP in the port that connects to the ISP
    - Switch(config)#**int f0/1**
    - Switch(config-if)#**no cdp enable**

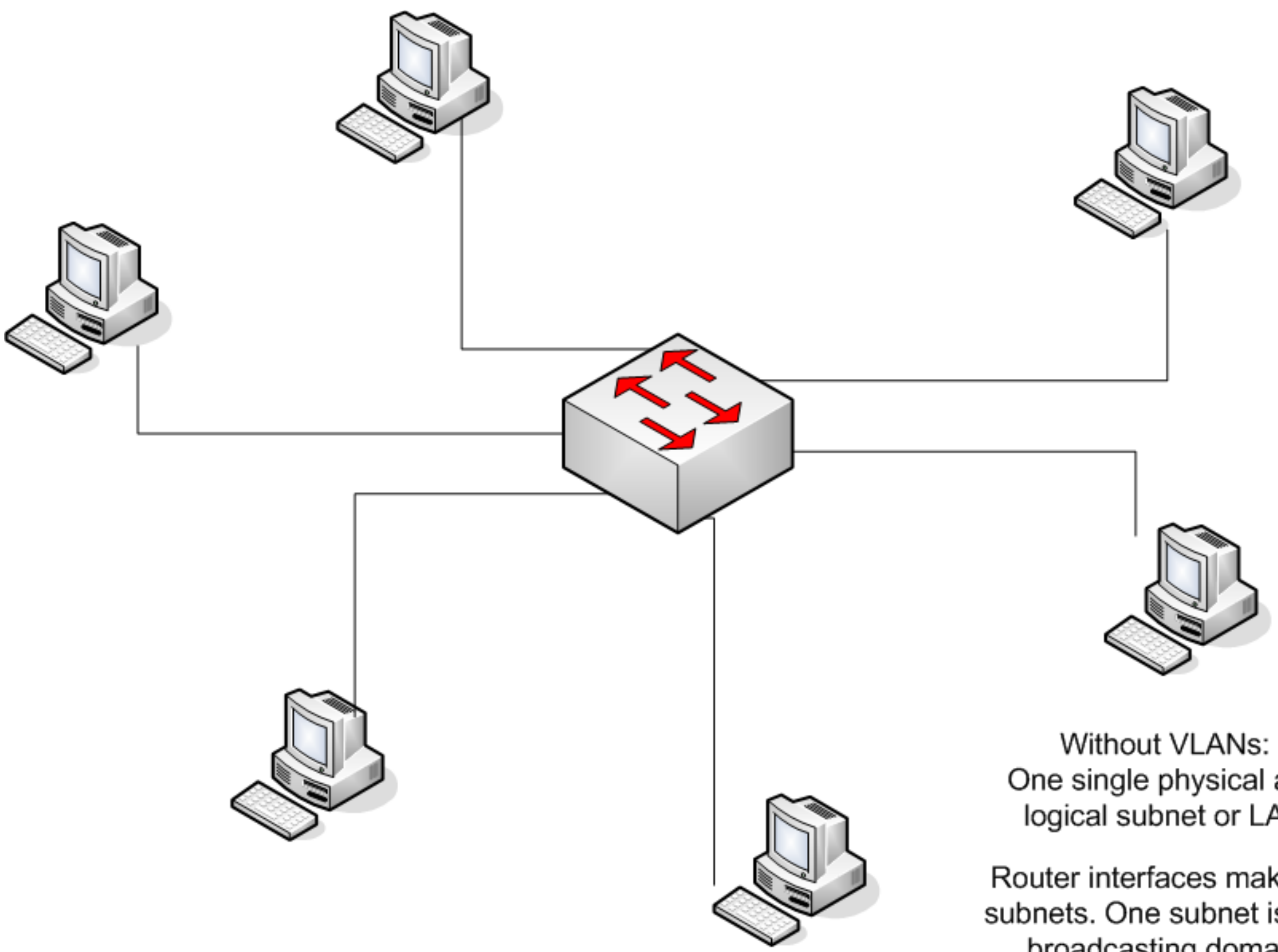
Basic troubleshooting (3/4)

- Ping
  - Executes from the User mode and from the Privileged Exec mode
  - Switch>**ping IP_ADDRESS_or_HOST_NAME**
    - !!!!!
      - 5 successful ICMP echo request/reply
    - …..
      - 5 unsuccessful replies
  - Also the extended ping exists in the Privileged Exec mode

Basic troubleshooting (4/4)

- Traceroute
  - Router>**traceroute IP_ADDRESS_or_HOST_NAME**
  - Also the extended traceroute exists in the Privilege Exec mode
  - Tracert is a windows command. It is not recognized in IOS but **trace** is, since it is unambiguous
- Telnet (or SSH)
  - If you can ping the destination but the telnet fails, you have a L7 issue
- Debug
  - Enables you to view events and problems in real time
  - Weakens the IOS performance because of the data retrieval
  - ”**debug all**” can halt the device operation and might crash it

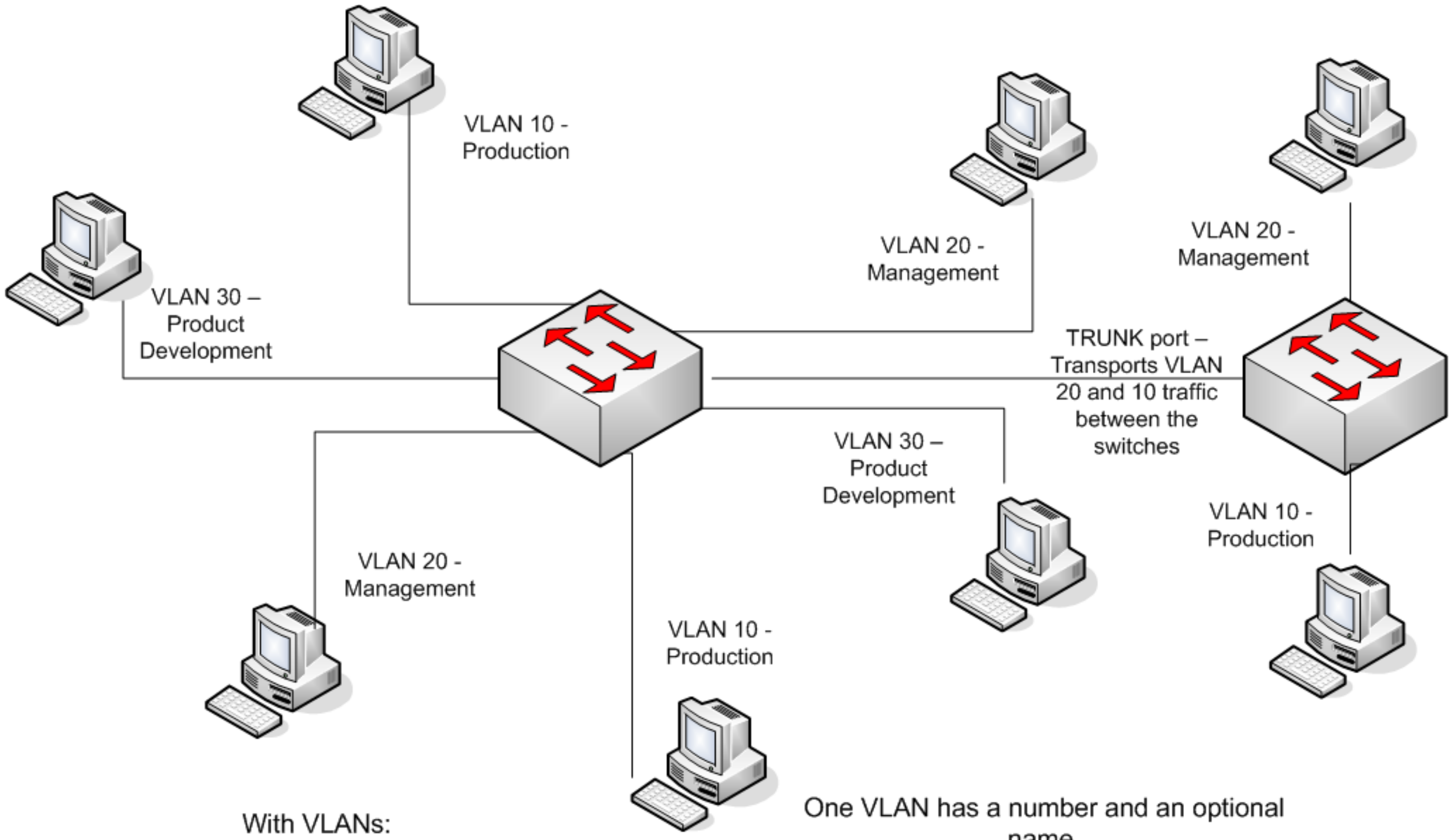
VLAN (Virtual LAN) and trunks (1/10)



Without VLANs:
One single physical and
logical subnet or LAN

Router interfaces make up
subnets. One subnet is one
broadcasting domain

VLAN (Virtual LAN) and trunks (2/10)



VLAN 10 - Production

VLAN 30 – Product Development

VLAN 20 - Management

VLAN 20 - Management

VLAN 20 - Management

TRUNK port – Transports VLAN 20 and 10 traffic between the switches

VLAN 30 – Product Development

VLAN 10 - Production

VLAN 10 - Production

With VLANs:
Several logical virtual LANs

One VLAN has a number and an optional name.
One VLAN is one broadcasting domain.
Note port types: access ports for hosts and trunk port for inter-switch connections

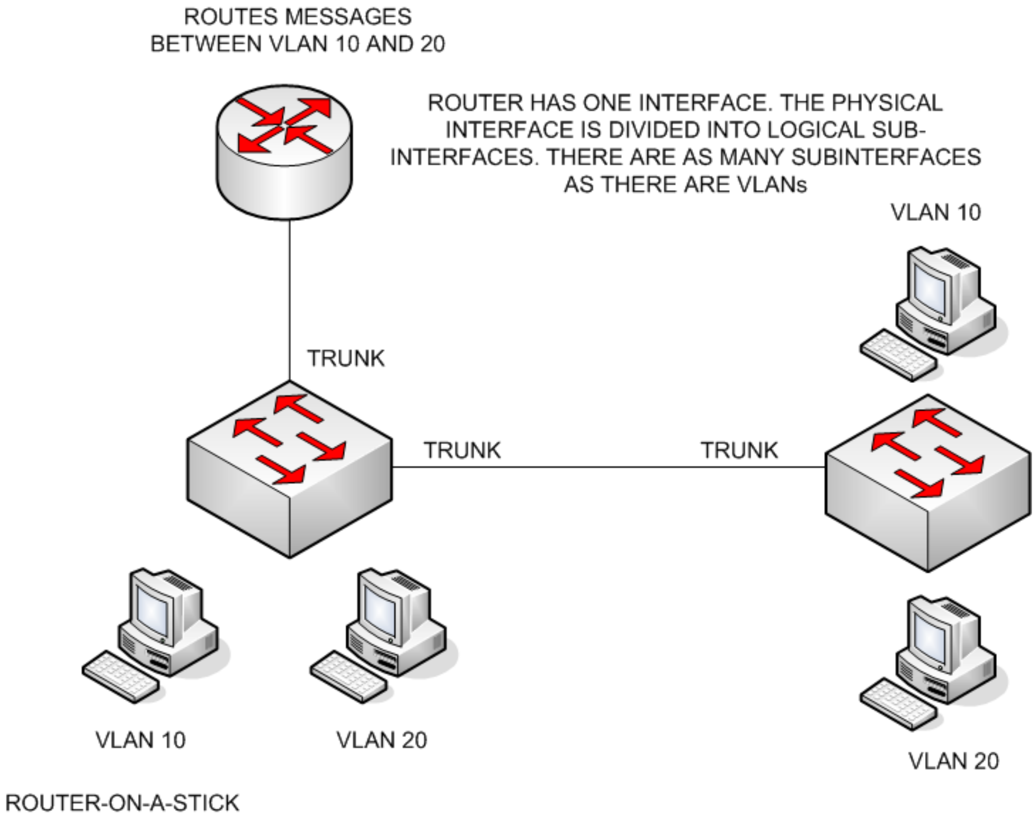
VLAN (Virtual LAN) and trunks (3/10)

- Without VLANs, you have LAN segments, that have all the hosts in the same logical group (subnet)
- Without VLANs, you must connect a host based on its physical location
- With VLANs, you can connect a host based on its logical group
- I.e. Production, Finance, Product Development, Management, etc.
- VLANs have a number and an optional name
- By default, all the switch ports are access ports and belong to VLAN 1. Also, by default, all the inter-switch communication occur in VLAN 1 (VTP messages, Cisco Discovery Protocol = CDP etc.). You should change this after you have come up with your VLAN scheme
- Connect all the managed switches into the same VLAN. That is, all the IP addresses of switches should belong to the same VLAN and subnet. Use i.e. VLAN 10 for management

VLAN (Virtual LAN) and trunks (4/10)

- To route inter-VLAN traffic, a router is needed
- Router-on-a-stick uses one router interface to route between VLANs (the switch port to connect to a router interface is configured as a trunk port)
  - Compare this to an old-fashioned way, where as many router ports were required as there are VLANs
  - Requires a support from a router
  - A router is configured using sub-interfaces (one sub-interface per VLAN)
  - Saves a lot of money (router interfaces are very expensive). So, use it
- There is one special VLAN, called the native VLAN. In a trunk port if an untagged frame is received, it is automatically assumed to belong to the native VLAN. By default, the native VLAN is VLAN 1. Hence, all the untagged (native Ethernet) traffic is propagated into VLAN 1 ports
- A native VLAN is born for example, when a hub with hosts on it is connected into the trunk connection between switches
- Even if possible, you should not change the number of the native VLAN (1)
- A native VLAN is dot1q concept

VLAN (Virtual LAN) and trunks (5/10)


ROUTES MESSAGES BETWEEN VLAN 10 AND 20
ROUTER HAS ONE INTERFACE. THE PHYSICAL INTERFACE IS DIVIDED INTO LOGICAL SUB-INTERFACES. THERE ARE AS MANY SUBINTERFACES AS THERE ARE VLANs
VLAN 10
TRUNK
TRUNK
TRUNK
VLAN 10
VLAN 20
VLAN 20
ROUTER-ON-A-STICK

VLAN (Virtual LAN) and trunks (6/10)

- Local broadcast traffic, such as ARP is not propagated outside the VLAN
- When the traffic is directed to outside a VLAN, the default IP gateway is used
- Logically speaking, VLANs are also subnets. However, VLAN is an L2 concept, whereas a subnet is an L3 concept
- A move of a user to a different location can be done provided that the old and new ports are connected to the same L2 network
- Security and broadcast containment for a single VLAN. Inter-VLAN traffic can have ACLs
- VLAN division can be logical, technical (i.e. VoIP in a single VLAN) , a security or a management issue (different QoS policies)

VLAN (Virtual LAN) and trunks (7/10)

- VLAN membership can be:
  - Static (fixed port configuration)
  - Dynamic, that is plug-and-play (automatic membership based on i.e. a MAC-address, a user name or a group (i.e. read from AD)). Requires a membership policy server (VMPS). Viable in larger organizations but requires more advanced and expensive switch models (such as 6500)
  - Voice (associated to VoIP phones)

VLAN (Virtual LAN) and trunks (8/10)

- Access ports
  - Port transports traffic belonging to a single VLAN
  - A standard Ethernet NIC
  - NIC does not need to understand any other traffic but IEEE 802.3 and Ethernet II frames
  - MY_SWITCH(config)#**int fast 0/1**
  - MY_SWITCH(config-if)#**switchport mode access**
  - MY_SWITCH(config-if)#**switchport access vlan 10**
  - MY_SWITCH(config-if)#**exit**
    - OR a group of interfaces at a time:
  - MY_SWITCH(config)#**int range fast 0/1-9**
  - MY_SWITCH(config-if-range)#**switchport mode access**
  - MY_SWITCH(config-if-range)#**switchport access vlan 10**
  - MY_SWITCH(config-if-range)#**exit**

VLAN (Virtual LAN) and trunks (9/10)

- Trunk ports
  - A trunk port is capable of carrying traffic for multiple VLANs
  - The original Ethernet frame must be modified to carry VLAN information
  - Requires a special NIC (and more expensive one. I.e. a server NIC can be configured to a trunk hence allowing access from different VLANs)
  - MY_SWITCH(config-if)#**switchport mode trunk**
  - Cisco supports two trunking methods:
    - ISL – Cisco proprietary
      - Being phased out by Cisco
      - MY_SWITCH(config-if)#**switchport trunk encapsulation isl**
    - IEEE 802.1Q – known as dot1q
      - The preferred method. Allows multiple vendors
      - MY_SWITCH(config-if)#**switchport trunk encapsulation dot1q**
    - Both ends must be configured to the same method
    - You can also configure a negotiation
      - MY_SWITCH(config-if)#**switchport trunk encapsulation negotiate**
    - Encapsulation options depend on the switch model (2950 & 2960 support only dot1q). Be sure to buy interoperable switches!

VLAN (Virtual LAN) and trunks (10/10)

- Trunk port configuration for:
  - Ports between switches
  - Ports between switches and routers
  - Server ports
- Trunk port can be configured to allow only certain VLANs
  - MY_SWITCH(config-if)#**switchport mode trunk**
  - MY_SWITCH(config-if)#**switchport trunk allowed vlan add 1,10,20**
  - It is easy to forget some VLANs from a trunk, so to make an administrators task easier, allow all:
  - MY_SWITCH(config-if)#**switchport trunk allowed vlan all**
  - If you allow all VLANs in a trunk, be sure to use VTP pruning
  - MY_SWITCH(config)#**vtp pruning**

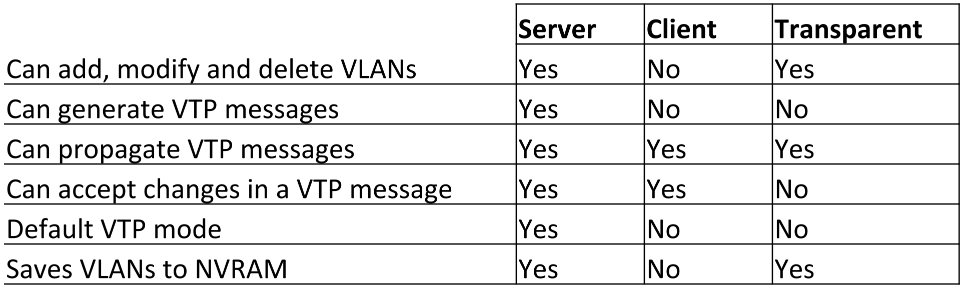
VLAN Trunk protocol – VTP (1/4)

- Cisco proprietary protocol to share VLAN info between Cisco switches (more specifically, between switch trunk ports)
- Removes much of the manual configuration of switches
- Compare two networks: first with 2 switches and second with 30 switches
  - You definitely need VTP, at least in larger networks
- Works only with switches, routers do not propagate VTP messages from one interface to another
- VTP domain
  - The VTP domain has identical info in all switches belonging to the domain
  - Compare to the Autonomous System in routing

VLAN Trunk protocol – VTP (2/4)

- Switch can be in one of the modes:
  - Client
  - Server
  - Transparent
- A trick that is sometimes used, is to hook up a new switch into the network in the client mode. Wait for it to receive VTP data and then configure it to a server. You need to configure, however, the VTP domain before that

VLAN Trunk protocol – VTP (3/4)



	Server	Client	Transparent
Can add, modify and delete VLANs	Yes	No	Yes
Can generate VTP messages	Yes	No	No
Can propagate VTP messages	Yes	Yes	Yes
Can accept changes in a VTP message	Yes	Yes	No
Default VTP mode	Yes	No	No
Saves VLANs to NVRAM	Yes	No	Yes

VLAN Trunk protocol – VTP (4/4)

- One option would be to configure one central switch to a server and all the rest to clients
- Worst-case scenario: You have a VLAN with 500 hosts and you happen to delete that VLAN in the server => All the switches drop this VLAN information
- Also, you can configure all or some of your switches to servers
- If you don't configure anything, the default mode is server
- To keep track of changes, the configuration revision number is incremented each time upon a change and an advertisement is made to other switches
- The highest configuration revision number is always the latest configuration. This revision is used and the other configurations get deleted
- VLAN database is saved in vlan.dat file, NOT in startup-config. "**erase startup-config**" does NOT delete vlan.dat. Hence, the worst-case scenario here is that when you boot up an erased switch into an existing network, it may have a higher config rev number (in vlan.dat) and overwrites the existing VLAN info
  - To be on a safe side, delete vlan.dat of a switch before connecting it to your network
  - MY_SWITCH#**delete vlan.dat**

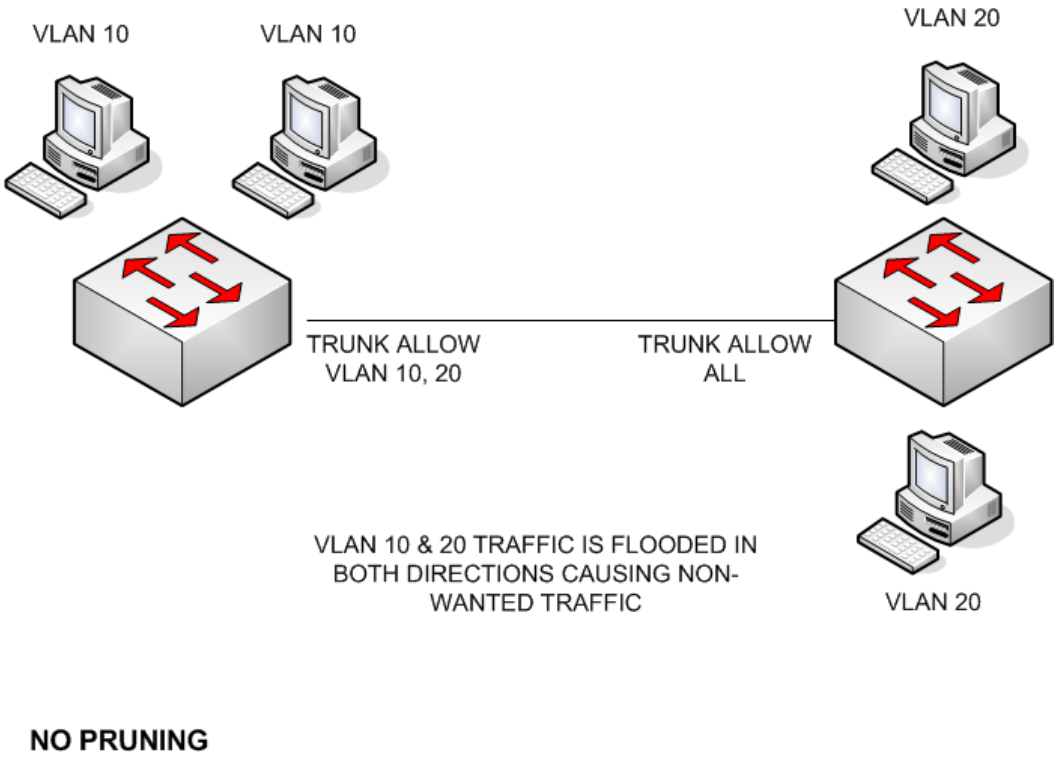
VTP Pruning (1/4)

- By default, any broadcast, multicast or an unknown unicast traffic is flooded out all ports associated the source VLAN, including trunks
- Flooding takes place into other switches, even if there are no active ports in them, as long as the trunk port has a source VLAN associated to it or all VLANs are associated to a trunk
- Think of a video multicast stream of 5 Mbps flooding everywhere
- Traffic rate can be downscaled by allowing manually only active VLANs in trunks but this is tedious. This is called static VLAN pruning. A better option would be using VTP Pruning and allowing all VLANs in trunk ports. This is called dynamic VLAN pruning

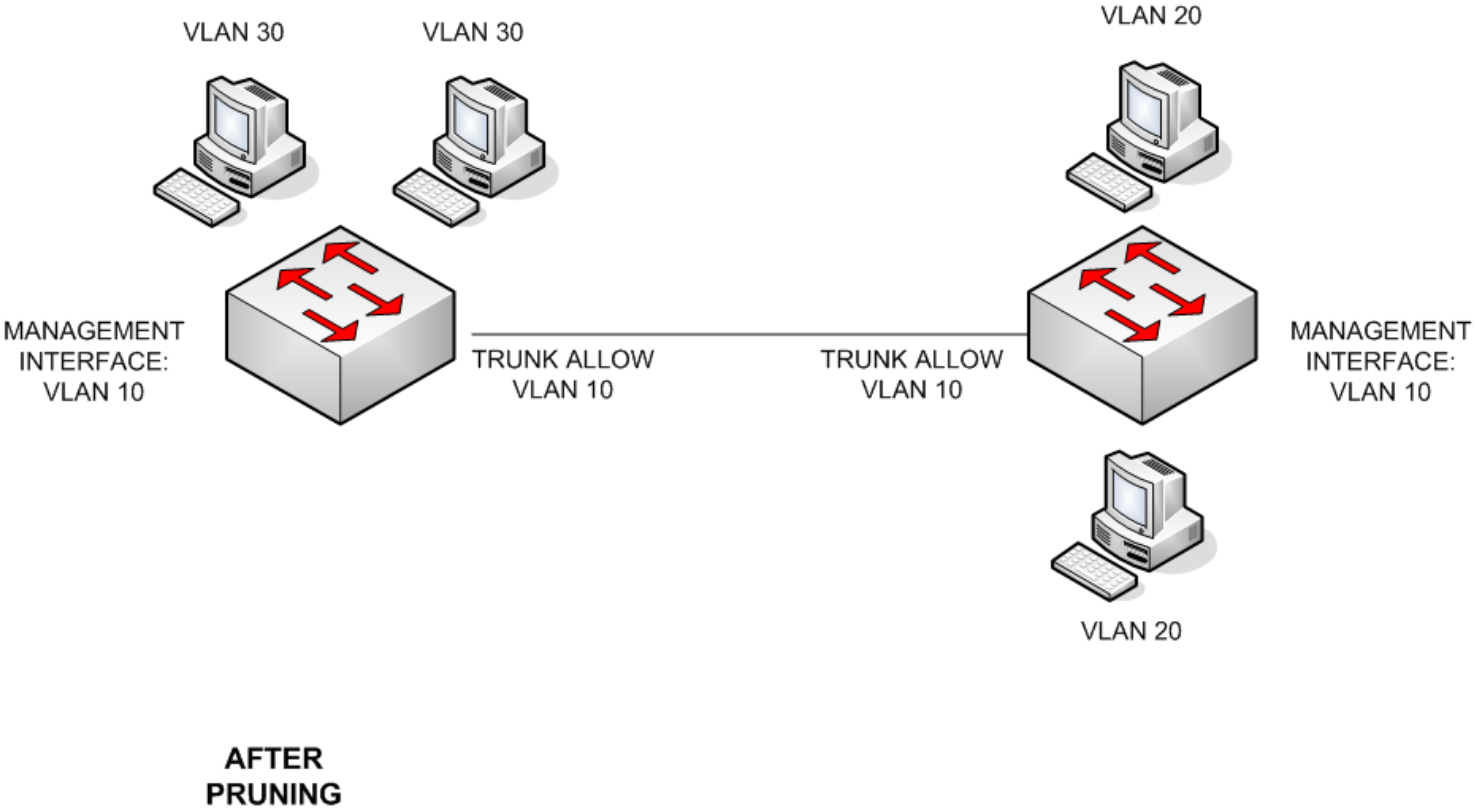
VTP Pruning (2/4)

- In order it to work, VTP pruning needs to be enabled only on one VTP server switch per VTP domain. All the switches involved in pruning are either VTP clients or VTP servers
- Transparent VTP switches must be pruned manually
- VTP pruning increases your network's efficiency, especially in larger networks, so you want to definitely enable it

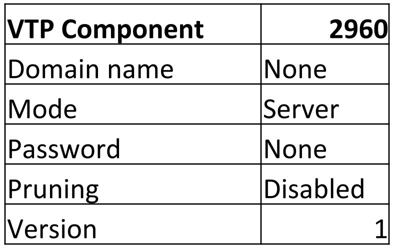
VTP Pruning (3/4)



VTP Pruning (4/4)



VTP configuration (1/2)



VTP Component	2960
Domain name	None
Mode	Server
Password	None
Pruning	Disabled
Version	1

VTP Default values

VTP configuration (2/2)

```
•   2960(config)#vtp domain MyDomain
•   2960(config)#vtp mode server|client|transparent
•   2960(config)#vtp password cisco
•   2960(config)#vtp pruning
•   2960#show vtp status
•   VTP Version                     : 2
•   Configuration Revision          : 1
•   Maximum VLANs supported locally : 64
•   Number of existing VLANs        : 5
•   VTP Operating Mode              : Server
•   VTP Domain Name                 :
•   VTP Pruning Mode                : Disabled
•   VTP V2 Mode                     : Disabled
•   VTP Traps Generation            : Disabled
•   MD5 digest                      : 0x70 0x01 0xF2 0x72 0x97 0xA1 0x35 0xEB
•   Configuration last modified by: 0.0.0.0 at 11-29-93 20:39:24
•   Local updater ID is 0.0.0.0 on interface Vl1 (lowest numbered VLAN interface
•   found)
•   2960#
```

- Obs! These commands do not work properly with the simulator!

Dynamic Trunk Protocol – DTP (1/3)

- Proprietary Cisco protocol, which dynamically forms and verifies trunk connections between switches
- Works only in some switch models
- If you can cope with ”**switchport mode trunk**” command – that is, statically define some ports to trunk, stick to it, because of its simplicity
- DTP has 5 modes

DTP (2/3)

DTP Mode	Generate DTP Messages	Default Frame Tagging
On or trunk	Yes	Yes
Desirable	Yes	No
Auto	No	No
Off	No	No
No-negotiate	No	Yes

On or Trunk – The connection is always assumed to be trunk.
The same behavior as: SWITCH(config-if)#**switchport mode trunk**

Desirable – The connection starts as an access link. If the remote end sends a DTP message and this is compatible, the trunk connection is formed

Auto – The same as Desirable, but does not send DTP messages. The connection starts as an access link. If the remote end sends a DTP message and this is compatible, the trunk connection is formed

No-negotiate – The interface is configured as a trunk connection, but DTP messages are not generated. This is used to connect with non-Cisco devices

No – The interface is configured as an access-link

Switch#**sh interfaces trunk**

DTP (3/3)

Your switch	Remote Switch
On	On, desirable, auto
Desirable	On, desirable, auto
Auto	On, desirable
No-negotiate	No-negotiate

The port configurations in order that the trunk connection is formed.

```
Switch(config-if)#switchport mode trunk
Switch(config-if)#switchport mode dynamic auto
Switch(config-if)#switchport mode dynamic desirable
Switch(config-if)#switchport nonegotiate
Switch(config-if)#no switchport mode trunk
```

VLAN creation (1/2)

- # of VLANs depend on the switch type
- Some VLANs are pre-configured in each switch:
  - VLAN 1 (native VLAN)
  - VLAN 1002-1005 – Token Ring and FDDI networks
  - To add or delete VLANs, you must be in a server mode (or in a transparent mode)
  - All the switch ports belong to VLAN 1 and are access ports by default. You should change this after your VLAN scheme
  - CDP, DTP and VTP are sent in VLAN 1
  - Before deleting a VLAN, reassign all the ports associated to a different VLAN

VLAN creation (2/2)

- Access ports:
  - Switch(config)#**int fast 0/1**
  - Switch(config-id)#**switchport mode access**
  - Switch(config-id)#**switchport access vlan VLAN#**
  - Switch(config-id)#**exit**
- VLAN creation (must be in a server mode (or transparent)):
  - Switch#**vlan VLAN#**
  - Switch(vlan)#**name Production**
  - Switch(vlan)#**exit**
- To verify the VLANs, type:
  - Switch#**sh vlan**
  - Switch#**sh int fast 0/1 switchport**
- If you see a line Access Mode VLAN: n (inactive), you have probably deleted a VLAN n
- Switch#**sh int trunk**
- Switch#**sh vtp**

Router-on-a-stick

- Router-on-a-stick is an L2/L3 configuration, where only one physical router interface is needed for inter-VLAN routing
- Router:
  - The physical interface of a router that is connected to a switch, is configured with no IP address (i.e. f0/0). In addition, the router's physical interface must be turned on with ”no shut”
  - Duplexing and speed are done on the physical interface, most other configs are done for a logical (subinterface) interface
  - Subinterfaces of a router are used to configure the settings for each VLAN, that is, the encapsulation mode (ISL or dot1q) and the IP address with a mask
  - The mask of an IP address is typed using dotted decimal notation i.e. 255.255.255.128. No CIDR notation is available with Cisco IOS (i.e. 192.168.1.1/24 is not allowed)
  - Use the subinterface numbers consistently:
    - VLAN 1 in the subinterface f0/0.1
    - VLAN 20 in the subinterface f0/0.20 etc.
- Switch:
  - VLANs used are added. VLAN 1 needs not to be added, because it exists by default
  - The switch port to connect to the router is configured as a trunk port
  - The switch ports to connect to hosts are configured to access ports
  - The management VLAN of the switch is configured and turned on with ”**no shut**”
- The hosts should be able to ping hosts in another VLANs. The traffic is routed through the router

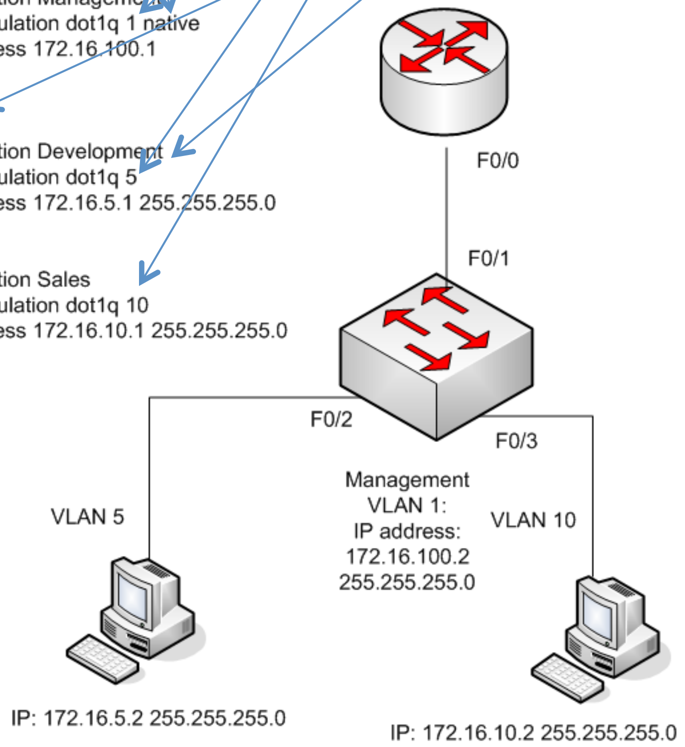
Router-on-a-stick

Native command does not work in a simulator

This is the VLAN id number

For consistency and ease of debugging, use the same subinterface number as VLAN id

```
Router# conf t
Router(config)# int f0/0
Router(config-if)# no ip address
Router(config-if)# no shut
Router(config-if)# int f0/0.1
Router(config-subif)# description Management
Router(config-subif)# encapsulation dot1q 1 native
Router(config-subif)# ip address 172.16.100.1
255.255.255.0
Router(config-subif)# exit
Router(config-if)# int f0/0.5
Router(config-subif)# description Development
Router(config-subif)# encapsulation dot1q 5
Router(config-subif)# ip address 172.16.5.1 255.255.255.0
Router(config-subif)# exit
Router(config)# int f0/0.10
Router(config-subif)# description Sales
Router(config-subif)# encapsulation dot1q 10
Router(config-subif)# ip address 172.16.10.1 255.255.255.0
Router(config-subif)# exit
Router(config)# exit
Router# copy run start
```



```
Switch# conf t
Switch(config)# vlan 5
Switch(config-vlan)# name Development
Switch(config-vlan)# exit
Switch(config)# vlan 10
Switch(config-vlan)# name Sales
Switch(config-vlan)# exit
Switch(config)# int f0/1
Switch(config-if)# switchport mode trunk
Switch(config-if)# exit
Switch(config)# int f0/2
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 5
Switch(config-if)# exit
Switch(config)# int f0/3
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 10
Switch(config-if)# exit
Switch(config)# int vlan 1
Switch(config-if)# ip address 172.16.100.2
255.255.255.0
Switch(config-if)# no shut
Switch(config-if)# exit
Switch(config)# ip default-gateway
172.16.100.1
Switch(config)# exit
Switch# copy run start
```

L2 redundancy

- Redundancy is implemented in different levels in larger networks
  - Several WAN links
  - L3 multipath and load-balance
  - L2 redundancy (Spanning Tree Protocol – STP, EtherChannels)
- Because of redundancy in L2, loops are created which can cause:
  - Multiple frame copies
  - Broadcast storms – remember, ARP uses local broadcast
  - Mislearning MAC addresses
- An efficient algorithm to prevent the loops is required
  - STP – Spanning tree protocol
  - Specifics are not included in this lab course
  - If you are interested, check:
  - http://en.wikipedia.org/wiki/Spanning_Tree_Protocol
  - Or better yet, check one of the course books

Private IP address spaces (inside NAT)

- Class A: 10.0.0.0 – 10.255.255.255
- Class B: 172.16.0.0 – 172.31.255.255
- Class C: 192.168.0.0 – 192.168.255.255
- The easiest to manage and remember is the 10.*.*.* address space, since it is the largest and simplest. That is why many organizations rely on this address space
- Even with IPv6 the private IP addresses will stick around

Routing basics (1/4)

- A router learns new routes either statically or dynamically
- Static route is learnt:
  - From directly connected interfaces
  - Manually configured (static) routes
  - A default route is used as a last resort
    - Common configuration in stub networks pointing to ISP router
    - IP address: 0.0.0.0
    - Subnet mask: 0.0.0.0
- Dynamic route is learnt by using a routing protocol:
  - RIPv1, RIPv2, IGRP+EIGRP (Cisco proprietary), OSPF, BGP, IS-IS
- Typically, a static routing or RIPv2 is used in simple and small networks. If your company policy is 100% Cisco, your choice of routing protocol could be EIGRP. If you are running a multivendor system, use OSPF or IS-IS
- Autonomous System is a group of routers under the same administration. AS's are numbered
- The routing protocols that understand AS are: EIGRP, OSPF, IS-IS and BGP. RIP does not
- A routing protocol within the AS is called IGP (Interior Gateway Protocol). EGP (Exterior Gateway Protocol) routes inter-AS traffic (BGP)

Routing basics (2/4)

- Cisco routers looks at two things when choosing the best route: Administrative Distance (AD) and routing metrics
- AD is a Cisco proprietary concept. A smaller AD number is preferable among multiple route informations. Obs! The same router can simultaneously run several routing protocols and can have static routes as well
- You can think of AD as how reliably the routing information was obtained
- If there are two or more routes to the same destination learnt from the same routing protocol, the routing metrics is used to pick up the best route. If also the routing metrics are the same, routing protocols use load-balancing (round robin)
- A routing metric can be: bandwidth, cost, delay, hop count, load, MTU or reliability depending on the routing protocol
- With point-to-point links, such as serial lines, a subnet consumes only 2 hosts. You would use a subnet mask 255.255.255.252 with p2p links, which gives a block size 4 and the number of hosts 2 (the first address is the network itself and the last address in the block is a local broadcast address)

Routing basics (3/4)

Administrative distance	Route type
0	Connected interface route
0 or 1	Static route
90	Internal EIGRP route (within the same AS)
110	OSPF route
120	RIPv1 and v2 route
170	External EIGRP (from another AS)
255	Unknown route (considered an invalid route)

If there are several learnt routes from different routing protocols, the Cisco router favors the route with the smallest AD

Routing basics (4/4)

- Three types of routing protocols:
  - Distance vector
  - Link state
  - Hybrid

Distance vector protocols (1/3)

- To find the path, the distance and direction are used
- Bellman-Ford algorithm
- Local and periodic broadcast messages
  - The complete routing table is broadcast to neighbors
- RIP, IGRP

Distance vector protocols (2/3)

- When a router receives a routing update from a neighbor, the following steps are performed (Bellman-Ford algorithm):
  1. Increment the metrics of the incoming routes. With RIP, this is a hop count
  2. Compare the network numbers
  3. If the advertised information from the neighbor is better, place to route to the routing table and remove the old entry
  4. If the advertised info is worse, ignore it
  5. If the advertised info already exists the same in the routing table, reset the entry timer
  6. If the advertised info has a different path and an equal metric as the old route information, both the routes are kept in the routing table. Some routing protocols support load balancing for equal-cost paths

Distance vector protocols (3/3)

- Disadvantages
  - Slow convergence time
  - Heavy network traffic
  - Routing loops may occur:
    - Split Horizon: Do not send the routing information back to the same direction you got it in the first place
    - Route poisoning: Advertise the failed link as unreachable (16 in RIP)
    - Holddown timers: Start a timer when the link goes up or down. The change is effective only after the timer has expired. This is especially useful with serial links, that might go up and down and up again (this is called flapping)
- Advantages
  - Easy to configure and debug
  - A low overhead for processing

Link state protocols (1/3)

- LSPs employ the Shortest Path First algorithm (SPF) invented by E.W. Dijkstra
- The whole network topology is stored in routers. This can lead to scalability problems with large networks
- The LSP routers create three separate tables:
  - Directly attached neighbors
  - Topology of the entire nw
  - Routing table
- OSPF and IS-IS
- OSPF is harder to setup and debug than RIP or EIGRP and IS-IS is even harder than OSPF

Link state protocols (2/3)

- LSPs use multicast to disseminate the nw information
- LSAs (Link State Advertisement) are multicast typically only in the event of topology changes
- LSAs are sent reliably. An acknowledgement is sent after receiving an LSA update
- Any time a change occurs, the routers run the SPF for their local topology table

Link state protocols (3/3)

- Advantages:
  - With the LSPs a hierarchical structure is employed. Hence, the changes in the nw do not necessarily propagate to all the routers. I.e. OSPF contain the propagations in areas
  - Multicast (vs. broadcast in distance vector protocols) reduces the overall processing and nw bandwidth
  - Typically in the boot-up the routers learn the nw topology and after that only the updates are sent out making this a more efficient protocol than distance vector protocols
  - LSPs support classless routing which allows you to use route summarization. A large group of contiguous routes can be summarized into smaller number of routes (VLSM, CIDR)
  - With route summarization and hierarchical routing, LSPs scale to larger nw sizes
- Disadvantages:
  - LSPs are CPU- and memory-intensive protocols
  - Running the SPF requires a lot more processing than incrementing and comparing the metrics of incoming routes with distance vector protocols

Hybrid protocols

- Typically, hybric protocols combine features from both the distance vector protocol and link state protocol
- EIGRP, BGP
- The most common IGP protocol today is OSPF followed by EIGRP. In SOHO networks, the static routes are the most typical implementation

Static routes (1/4)

- A static route is a manually configured route in your router
- A typical way to handle routes in small networks
- Not a viable way in medium to large networks or when you have many subnets in your nw

Static routes (2/4)

- Static route configuration:
  - Router(config)#**ip route DESTINATION_NW [subnet_mask] IP_ADDRESS_OF_NEXT_HOP_NEIGHBOR [administrative distance] [permanent]**
  - OR
  - Router(config)#**ip route DESTINATION_NW [subnet_mask] EXIT_INTERFACE [administrative_distance] [permanent]**
  - The first parameter is the destination network number followed by a subnet mask. If the subnet mask is left out, the router uses default masks depending on the destination nw. A nw – 255.0.0.0, B nw – 255.255.0.0, C nw – 255.255.255.0
  - You can configure the next hop address by the IP address or by the exit interface
    - If the outgoing link is a multi-access link, use the IP address. The default AD will be 1
    - If the outgoing link is i.e. a point-to-point link, you can use the next hop interface. The default AD will be 0
  - Router(config)#**ip route 172.16.10.1 255.255.255.0 10.0.1.1**
  - Router(config)#**ip route 172.16.10.1 255.255.255.0 serial0**
  - Remember the administrative distance from a previously shown slide. If you are going to use both the static addresses and dynamic routing in parallel, you could set the AD of static routes to 200, which is greater than the AD of any routing protocol. The static routes would become backup routes in case the dynamic routing fails. If you have redundancy in your networks, you could set the AD of secondary static routes to 201 etc.
  - Administrative distance is not supported by the simulator

Static routes (3/4)

- Static route configuration (cont.):
  - By default, the next hop address as an IP address has an administrative distance of 1 and the next hop address as an outgoing interface has an administrative distance of 0
    - You can change the default values with the administrative distance parameter
  - You can create multiple static routes to the same destination. You need this when you use a primary route and a secondary route. If the primary route fails, the secondary route is used. Then you must use a different administrative distance value, i.e. a default value for the primary route and the value 2 for the secondary router
  - If you omit the permanent parameter, the router drops the route from its memory in case the link goes down. You might want to use the permanent parameter if you never want the packets to follow a specific route, perhaps due to security reasons

Static routes (4/4)

- A default route can be configured to enable the last resort. If the router does not know the destination, it propagates the packets to the default route
- The default route is typically used in stub networks with only one exit path
- Router(config)#**ip route 0.0.0.0 0.0.0.0 IP_ADDRESS_OF_NEXT_HOP_NEIGHBOR [administrative_distance] [permanent]**
- OR
- Router(config)#**ip route 0.0.0.0 0.0.0.0 INTERFACE_TO_EXIT [administrative_distance] [permanent]**
- The network 0.0.0.0 represents all the networks and the mask 0.0.0.0 represents all the hosts in the specified network
- If you use the defaut route, you must also give a specific command:
  - Router(config)#**ip classless**
- You can also apply the default network:
  - Router(config)#**ip default-network 192.168.1.0**
- To verify the static and default routes:
  - Router#**sh ip route**
  - Output: C – connected, S – static, * - default route

Classful and classless routing protocols

- Classful protocols:
  - RIPv1, IGRP (no longer supported by Cisco)
- Classless protocols:
  - RIPv2, OSPF, EIGRP, IS-IS, BGP
- When a classful router advertises a route, it does not contain the subnet mask. You can only use one subnet mask for all the devices in the network (VLSM is not applicable)
- A classless router advertises the subnet mask, too, making it possible to use VLSM and route-summarization

RIP (1/2)

- Maximum hop count is 15 making this protocol suitable only for small networks. The hop count 16 means unreachable
- RIP sends the complete routing table to all active interfaces every 30 seconds
- Works well in small networks
- RIPv1 and RIPv2 differences:
  - RIPv1 is classful, whereas RIPv2 is classless
  - RIPv1 uses broadcast, whereas RIPv2 uses multicast
  - Only RIPv2 supports VLSM
  - Authentication is an option with RIPv2
  - RIPv2 supports discontiguous networks (see http://answers.yahoo.com/question/index?qid=20080722132707AAn9sit)
- To configure RIP, just turn on the protocol with ”router rip” command and tell the RIP all the networks that are advertised with ”network” command. Typically, you want to advertise all the directly connected networks. To enable RIPv2, use the ”version 2” command
  - Router(config)#**router rip**
  - Router(config-router)#**version 2|1**
  - Router(config-router)#**network A.B.C.D**
  - Router(config-router)#**exit**
- RIPv1 and RIPv2 are configured as classful, meaning that only the base class network is typed. For example, if you have the subnets 172.16.1.0 and 172.16.2.0 you only configure one classful network 172.16.0.0. The RIPv2 router finds the subnets and the masks from the interface settings. With 10.0.1.0 and 10.0.2.0 you would supply the network 10.0.0.0 only
- You can turn off sending RIP updates out to the interface. You might want to do this with the interfaces, that have no routers on the other end
  - Router(config-router)#**passive-interface s0/0/0**
- To disable the auto-summary:
  - Router(config-router)#**no auto-summary**

RIP (2/2)

- RIP uses 4 timers to adjust its performance
  - Route update timer: The interval between periodic routing updates (typically 30s)
  - Route invalid timer: The time that elapses before the route is considered invalid (typically 180s)
  - Hold-down timer: The time during which the routing information is suppressed. Routers enter into the holddown state when an update packet is received indicating that the route is unreachable
  - Router flush timer: The time between a route comes invalid and its dropping from the routing table (typically 240s). Must be greater than route invalid timer
- Router(config-router)#**timers basic 30 90 180 270 360**
  - 30 = Update timer
  - 90 = Invalid timer
  - 180 = Hold-down timer
  - 270 = Flush timer
  - 360 = Sleep time

ACL (1/24)

- Access Control Lists
- Creating, updating and debugging ACL can be pretty cumbersome
- Cisco ACL supports in addition to IP also IPX, AppleTalk and others
- Filters traffic upon either leaving (outbound) or entering (inbound) the interface
- For inbound ACL, the ACL is processed before any further processing by IOS. With the outbound ACL, the packet is first routed to the interface and then the outbound ACL is processed
- ACL applied to outbound cannot filter traffic that originates from the IOS itself
- Applied to:
  - Remote access (Telnet, SSH)
  - Routing info
  - Traffic prioritizing with queues
  - DDR triggering (Dial-on-demand)
  - IPSec VPN
  - And many others

ACL (2/24)

- ACL commands either permit or deny traffic
- Types of ACL:
  - #1 numbered and named
  - #2 standard and extended
    - Standard IP ACLs filter on the source IP address inside a packet
    - Extended IP ACLs filter on the source and destination IP address in the packet, the protocol (TCP, UDP, ICMP and so on) and protocol info (such as TCP or UDP port numbers or ICMP message type)

ACL (3/24)

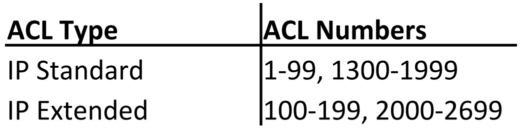
Filtered Information	Standard IP ACL	Extended IP ACL
Source address	Yes	Yes
Destination address	No	Yes
Protocol (TCP, UDP etc.)	No	Yes
Protocol information (i.e. port number)	No	Yes

ACL (4/24)

- ACL is a list of statements that are linked together with a name or a number
- ACL is processed top-down. A packet is compared with one statement at a time (permit or deny). A new entry is added to the bottom of the list
- You should always put the most restricting statements in the top of the list and the least restricting to the bottom of the list
- The following order is applied with ACLs:
  - #1 Once a match (permit or deny) is found, no further statements are processed
  - #2 The order of the statements is important, because once a match is found, the rest of the statements are ignored
  - #3 If no match is found in the list, the packet is dropped (implicit deny)
- ACL should include at least one permit statement, otherwise it is always dropped
- Only one IP ACL can be applied to an interface in each direction (inbound and outbound)
- If you apply an empty ACL to an interface, it permits all the traffic (implicit deny requires at least one permit or deny statement). But you should never use empty ACLs
- You cannot remove one line from an access list. Trying this will result in the removal of the entire list
- Place IP standard access lists as close to the destination as possible
- Place IP extended access lists as close to the source as possíble

ACL (5/24)

- To create a numbered ACL:
  - Router(config)#**access-list ACL# permit|deny conditions**
  - You can only use a prespecified ACL number as follows:



ACL Type	ACL Numbers
IP Standard	1-99, 1300-1999
IP Extended	100-199, 2000-2699

  - On the other hand, you can use named ACLs as much as you have memory in your router
- To apply the ACL into the interface, you must activate it:
  - Router(config)#**int type [slot#]/port#**
  - Router(config-if)#**ip access-group ACL# in|out**
- You can apply ACL also into the subinterfaces, i.e. f0/0.1

ACL (6/24)

- Instead of typing all the possible IP addresses, you can apply wildcard masks to match a range of addresses
- Do not confuse with subnet masks!
- A wildcard mask is 32 bits long as is the subnet but the bits are interpreted differently

ACL (7/24)

Bit Value	**Subnet mask**	**Wildcard Mask**
0	Host component	Must match
1	Network component	Ignore

Wildcards:

0 in a bit position means that the corresponding bit in the address of the ACL statement must match the same bit in the IP address of the examined packet

1 in a bit position means that the corresponding bit in the address of the ACL statement does not have to match the same bit in the IP address of the examined packet

ACL (8/24)

- A wildcard mask is actually an inverted subnet mask
- An example: a subnet mask 255.255.0.0
  - In a binary format this is: 11111111.11111111.00000000.00000000
  - And the mask after inverting is: 00000000.00000000.11111111.11111111
  - Which is: 0.0.255.255
  - This wildcard mask means that the first 16 bits of the IP address of the packet must match with the 16 first bits of the ACL statement
  - Inverting bytes 0 and 255 is easy. 0 becomes 255 and 255 becomes 0
  - An easy way to invert a byte from a subnet mask to a wildcard is to subtract the corresponding byte from 255. What becomes of the subnet mask 255.255.240.0?
    - A: 255-255=0
    - B: 255-255=0
    - C: 255-240=15
    - D: 255-0=255
    - 255.255.240.0 -> 0.0.15.255
  - You can use an online tool for this:
    - http://www.subnet-calculator.com/
- Two special case wildcards: 0.0.0.0 and 255.255.255.255
  - 0.0.0.0 means that all the 32 bits of the address in the ACL statement must match the IP address of an examined packet. This mask is called a host mask. IOS converts this mask to ”host”
  - 172.16.1.2 0.0.0.0 means that the address must be exactly 172.16.1.2. Alternatively, you can type: ”host 172.16.1.2”
  - 0.0.0.0 255.255.255.255 means any address will do. IOS converts this mask to ”any”. Alternatively, you can type: ”any”

ACL (9/24)

Examples:

IP Address	Wildcard Mask	Matches
0.0.0.0	255.255.255.255	Matches with any address
10.0.1.1	0.0.0.0	Matches only when the address is 10.0.1.1
10.0.1.0	0.0.0.255	Matches when the address matches the subnet 10.0.1.0/24
		10.0.1.0-10.0.1.255
10.0.2.0	0.0.1.255	Matches when the address matches the subnet 10.0.2.0/23
		10.0.2.0-10.0.3.255
10.1.0.0	0.0.255.255	Matches when the address matches the subnet 10.1.0.0/16
		10.1.0.0-10.1.255.255

ACL (10/24)

- Standard numbered ACLs:
  - Router(config)#**access-list 1-99|1300-1999 permit|deny source_IP_address [wildcard mask] [log]**
  - The condition is based only on the source IP address
  - If you omit the wildcard mask, it defaults to 0.0.0.0 requiring an exact match. Obs! The simulator behaves differently. You need to type a mask (be it 0.0.0.0) or type: ”host IP_address”
  - You can use as a source IP-address a parameter ”**host IP_address**” (same as IP_address 0.0.0.0) or ”**any**” (same as 0.0.0.0 255.255.255.255)
  - log causes the match to be printed on the console port
- Example:
  - Router(config)#**access-list 10 deny host 10.0.1.2** – OR - Router(config)#**access-list 10 deny 10.0.1.2 0.0.0.0**
  - Router(config)#**access-list 10 permit any**
  - Router(config)#**int serial 0**
  - Router(config-if)#**ip access-group 10 in**
- The access-list 10 is processed as follows:
  - The access-list is applied to the inbound traffic of serial line 0
  - First, deny the traffic, if the source IP address is 10.0.1.2
  - If the source IP address was not 10.0.1.2, permit all the traffic
- Example:
  - Router(config)#**access-list 20 deny 10.0.1.0 0.0.0.255**
  - Router(config)#**access-list 20 permit any**
  - Router(config)#**int serial 1**
  - Router(config-if)#**ip access-group 20 out**
- The access-list 20 is processed as follows:
  - The access-list is applied to the outbound traffic of the serial line 1
  - First, deny the traffic from the subnet 10.0.1.0/24 or 10.0.1.0-10.0.1.255
  - If the source address was not 10.0.1.0/24, allow all the traffic

ACL (11/24)

- You definitely want to restrict the VTY access (Telnet, SSH) to your router
- You apply the ACL to the VTYs, not the router interfaces as follows:
  - Router(config)#**access-list 20 permit 192.168.1.0 0.0.0.255**
  - Router(config)#**line vty 0 4**
  - Router(config-line)#**access-class 20 in**
- This ACL allows the VTY only from the subnet 192.168.1.0/24
- To secure VTY, you need to apply a login and a password plus a standard ACL allowing the traffic only from the management stations
- Telnet is clear-text and IP addresses can be spoofed. So, better to use SSH

ACL (12/24)

- Extended ACL can match on the following info:
  - Source and destination IP address
  - TCP/IP protocol (IP, TCP, UDP, ICMP etc)
  - Protocol info, such as port numbers or message types

ACL (13/24)

- Router(config)#**access-list 100-199|2000-2699 permit|deny IP_protocol source_address source_wildcard_mask [protocol_information] destination_address destination_wildcard_mask [protocol_information] [log]**
- The most typical IP_protocol values are: ip, icmp, tcp, gre, udp, igrp, eigrp, igmp, ipinip, nos and ospf
  - If you want to match on any IP protocol, use the keyword ip
  - If you want to filter by application layer protocol, you have to choose the appropriate layer 4 transport protocol. For example, to filter ftp you choose TCP
  - You can use ”**host IP_address**” (same as IP_address 0.0.0.0) or ”**any**” (same as 0.0.0.0 255.255.255.255) parameters for both source and destination
- You must specify both the source and destination addresses and their wildcard masks
- Log has the same meaning as with a standard ACL
- After creating the ACL, you must activate it to the router’s interface with ”**ip access-group**” command, same as the standard ACL

ACL (14/24)

- To configure ACL for TCP or UCP use the following:
  - Router(config)#**access-list 100-199|2000-2699 permit|deny tcp|udp source_address source_wildcard_mask [operator source_port_#] destination_address destination_wildcard_mask [operator destination_port_#] [established[[log]**
  - With TCP and UDP you can specify the port numbers or names of the source, destination or both
    - You specify an operator to define how to make a match with numbers of names
    - TCP and UDP operators are as follows:

– lt	- less than
– gt	- greater than
– neq	- not equal to
– eq	- equal to
– range	- range of port numbers

ACL (15/24)

- With TCP and UDP connections, you list either a name or a number of the port
  - i.e. tfpt – 69, ssh - 22
- The names and numbers can be found:
  - http://www.iana.org/assignments/port-numbers
- The established keyword is used only with TCP
  - TCP traffic that originates from your network and is connected to outside
  - With established keyword you either deny or allow returning TCP traffic, that has a certain flag set in the TCP segment

ACL (16/24)

- With ICMP you use:
- Router(config)#**access-list 100-199|2000-2699 permit|deny icmp source_address source_wildcard_mask destination_address destination_wildcard_mask [icmp_message] [log]**
- ICMP does not use ports, only messages (it is L3 protocol)
- Omitting the ICMP message type, all message types are included
- The most common ICMP messages are:
  - echo - used by ping
  - echo-reply - reply to ping
  - host-unreachable - subnet is reachable, but the host is not responding
  - net-unreachable - network/subnet is not reachable

ACL (17/24)

- To remove ACL from an interface:
  - Router(config)#**int serial 0**
  - Router(config-if)#**no ip access-group ACL_#**
- ACL is deleted by:
  - Router(config)#**no access-list ACL_#**
- To view the ACLs:
  - Router#**sh access-lists**
- To view a specific ACL number:
  - Router#**sh access-lists 100**
- To view the interfaces, that have ACLs hooked:
  - Router#**sh ip interface**
- To view the ACLs and the interfaces:
  - Router#**sh run**

ACL (18/24)

- Extended ACL examples:
- Router(config)#**access-list 100 permit udp any host 10.0.1.2 eq dns log**
- Router(config)#**access-list 100 permit tcp 10.0.0.0 0.0.255.255 host 10.0.1.3 eq telnet log**
- Router(config)#**access-list 100 permit icmp any 10.0.0.0 0.0.255.255 echo-reply log**
- Router(config)#**access-list 100 deny ip any any log**
- Router(config)#**int ethernet 0**
- Router(config-if)#**ip access-group 100 in**
- The ACL filters the inbound traffic of the ethernet 0 interface. This ACL permits DNS for the host 10.0.1.2 from any host, permits the telnet traffic to the host 10.0.1.3 from the 10.0.*.* address space, permits echo-reply from any host to the subnet 10.0.0.0/16 and denies all the rest of the traffic
- ”Router(config)#**access-list 100 deny ip any any log**” could be left off, but this line forces all the dropped traffic to be printed on the console port (which implicit deny does not do)

ACL (19/24)

- Example:
- Router(config)#**access-list 110 deny tcp any host 10.0.1.2 eq 23 log**
- The ACL 110 denies telnet traffic from any host that have a destination 10.0.1.2. The matches are printed on the console port
- Router(config)#**access-list 120 deny tcp any 172.16.1.0 0.0.0.255 eq 23**
- Router(config)#**access-list 120 deny tcp any 172.16.2.0 0.0.0.255 eq 23**
- Router(config)#**access-list 120 permit ip any any**
- Router(config)#**int f0/0**
- Router(config-if)#**ip access-group 120 out**
- Router(config-if)#**exit**
- Router(config)#**int f0/1**
- Router(config-if)#**ip access-group 120 out**
- The ACL 120 is applied to two fastethernet interfaces, outbound traffic. Telnet is denied from any host to the subnet 172.16.1.0/24 and 172.16.2.0/24. All the rest of the traffic is allowed

ACL (20/24)

- Named ACLs
- You can delete a single entry in a named ACL without deleting the whole ACL
- Router(config)#**ip access-list standard|extended ACL_name**
- This command takes you into a subconfiguration mode:
- Router(config-std-acl)# - OR – Router(config-ext-acl)#
- With a standard ACL, use:
  - Router(config)#**ip access-list standard ACL_name**
  - Router(config-std-acl)#**permit|deny source_IP_address [wildcard_mask]**
- With an extended ACL, use:
  - Router(config)#**ip access-list extended ACL_name**
  - Router(config-ext-acl)#**permit|deny IP_protocol source_IP_address wildcard_mask [protocol_information] destination_IP_address wildcard_mask [protocol_information] [log]**

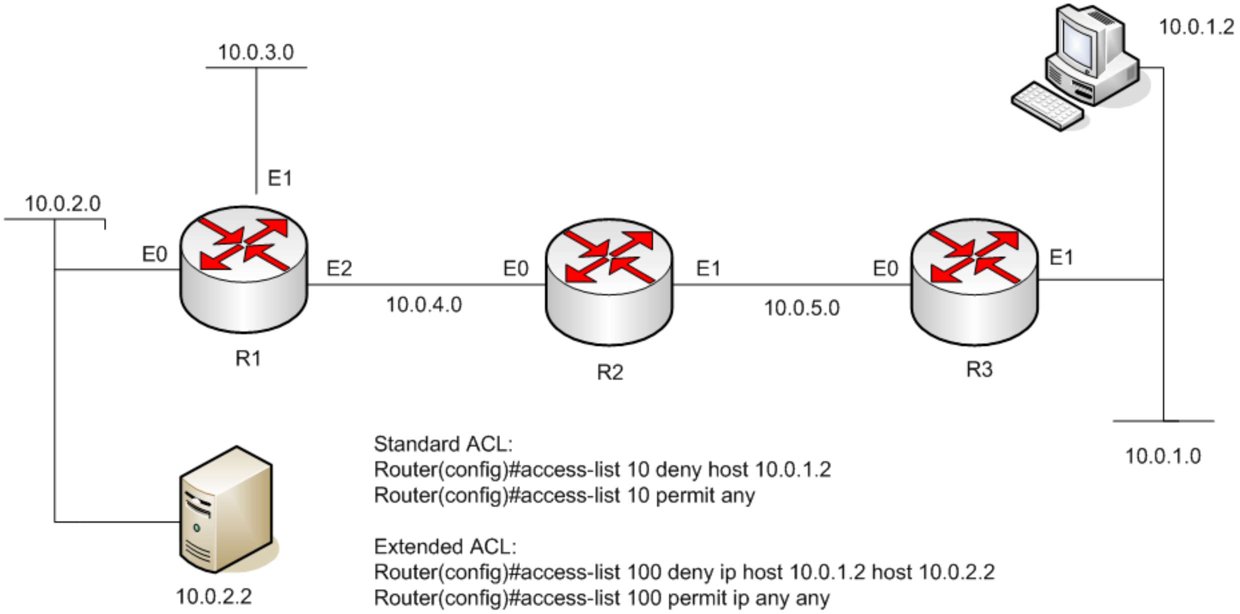
ACL (21/24)

- You can insert comments to your ACLs:
  - With a numbered ACL:
    - Router(config)#**access-list ACL_# remark Your_remark**
  - With a named ACL:
    - Router(config)#**ip access-list standard|extended ACL_name**
    - Router(config-{std|ext}-acl)#**remark Your_remark**

ACL (22/24)

- Starting from IOS 12.3 you can edit individual entry lines in your ACL. This is called a sequenced ACL
- With the sequenced ACL, each ACL command is given a sequence number starting from 10 and having an increment of 10
- You can view the sequense numbers by:
  - Router#**sh access-list**

ACL (23/24)



Standard ACL:

```
Router(config)#access-list 10 deny host 10.0.1.2
Router(config)#access-list 10 permit any
```

Extended ACL:

```
Router(config)#access-list 100 deny ip host 10.0.1.2 host 10.0.2.2
Router(config)#access-list 100 permit ip any any
```

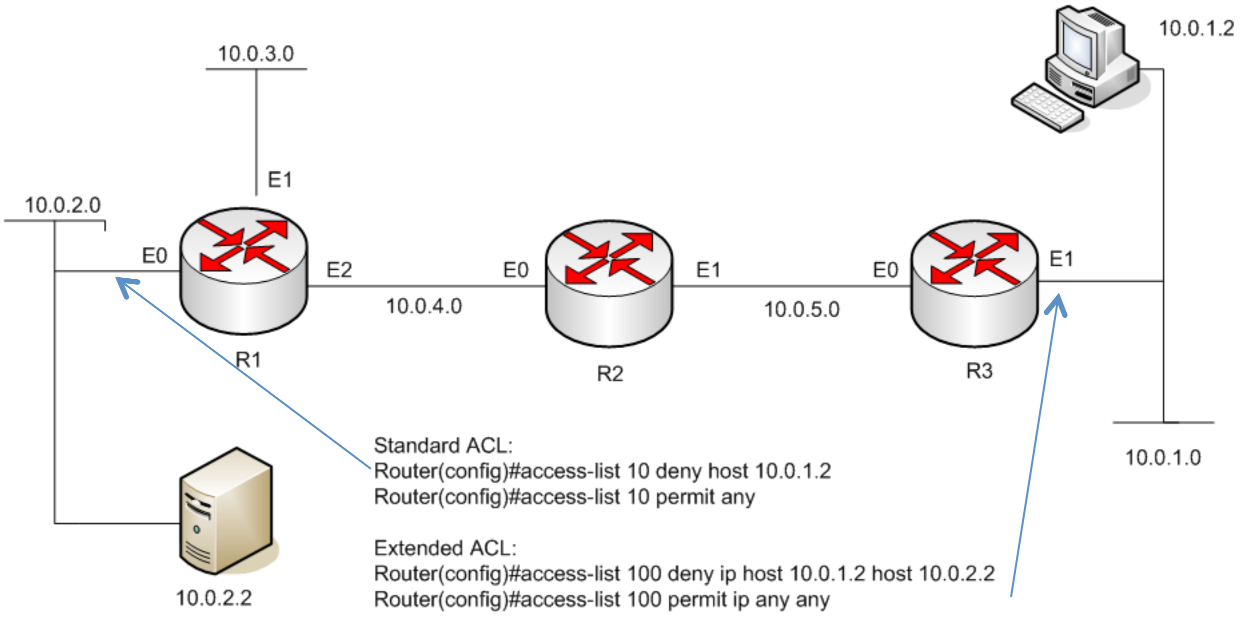
There are two rules about placement of standard and extended ACLs in your network:

#1 Standard ACLs should be placed as close to the destination as possible
#2 Extended ACLs should be placed as close to the source as possible

You want to deny the access from the host 10.0.1.2 to the server 10.0.2.2.

Q: Now, where would you place these two ACLs in the network?

ACL (24/24)



Standard ACL:

```
Router(config)#access-list 10 deny host 10.0.1.2
Router(config)#access-list 10 permit any
```

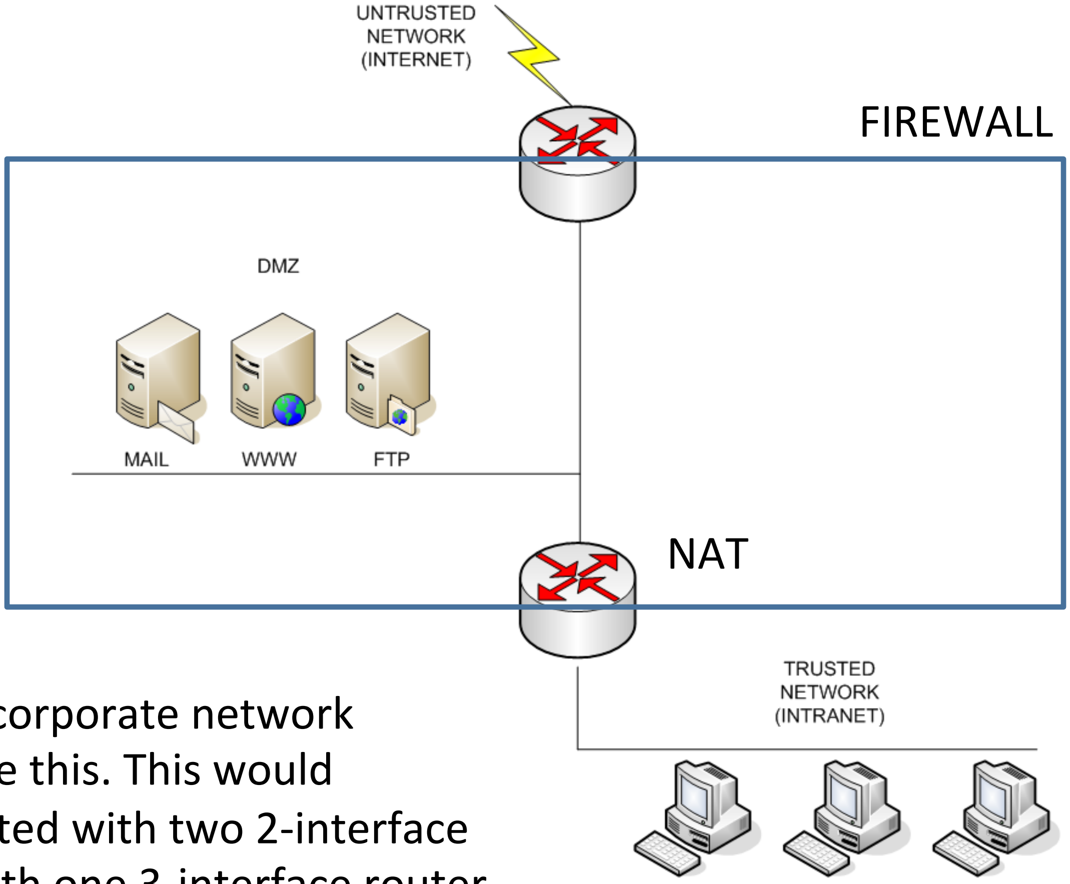
Extended ACL:

```
Router(config)#access-list 100 deny ip host 10.0.1.2 host 10.0.2.2
Router(config)#access-list 100 permit ip any any
```

The standard ACL goes to the E0 out interface of R1.
The extended ACL goes to the E1 in interface of R3.

Why?

NAT (1/13)



The modern corporate network could look like this. This would be implemented with two 2-interface routers or with one 3-interface router.

NAT (2/13)

- Network Address Transtation (NAT) addresses three problems:
  - #1 shortage of public IPv4 addresses
  - #2 hides the implementation of your inner network
  - #3 makes you independent of your ISP
- NAT translation is done in a router, a firewall or a server
- With NAT, you can have overlapping internal addresses in for example in your branch offices
- Address translation variations:
  - #1 Static address translation
  - #2 Dynamic address translation
  - #3 Port address translation (PAT)
- Private IP address spaces (inside NAT)

– 1 A-class address:	10.0.0.0 – 10.255.255.255
– 16 B-class addresses:	172.16.0.0 – 172.31.255.255
– 256 C-class addresses:	192.168.0.0 – 192.168.255.255

NAT (3/13)

Term	Definition
Inside	Addresses located inside of your nw
Outside	Addresses located outside of your nw
Local	The IP address physically attached to a device
Global	The public IP address physically or logically attached to a device
Inside local IP address	A device inside with a private IP address
Inside global IP address	A device inside with a public IP address
Outside global IP address	A device outside with a public IP address
Outside local IP address	A device outside with a private IP address

NAT terminology

NAT (4/13)

Translation type	Meaning
Static	A manual IP address (and possibly port number) translation is done between inside and outside devices
Dynamic	An automatic IP address (and possibly port number) translation is done between inside and outside devices
Port address translation (PAT)	All the inside IP addresses are mapped to a single outside public IP address. The unique port number (TCP, UDP) is used to distinguish between the inside devices

NAT (5/13)

- Static NAT
  - The NAT box translates the IP addresses from the outside into inside addresses (one-to-one mapping)
  - The NAT box translates the IP addresses from the inside into outside addresses (one-to-one mapping)
  - Typically, static NAT is used for the traffic that originates from the outside such as accessing a Web Server in your corporate network
  - You have N public IP addresses and N private IP addresses that are mapped statically by the administrator
  - Note, that no port numbers are used

NAT (6/13)

- Dynamic NAT
  - Performs better than static NAT, especially if you translate traffic that comes from the inside
  - An IP address pool of public IP addresses is recycled with that originates from the inside
  - Typically, used for the traffic that originates from the inside
  - Usually the pool of public IP addresses is smaller than the private IP addresses that can generate traffic
  - Note, that no port numbers are used

NAT (7/13)

- Port address translation (PAT)
  - PAT is heavily used today
  - One public IP address is overloaded. The TCP/UDP port numbers are used to distinguish between connections
  - PAT is used when you have less public IP addresses than you have inside devices needing to get outside
  - The following concepts are used with PAT:
    - Inside local IP address – original source inside IP address
    - Inside local port number – original source port number
    - Inside global IP address – public source IP address after translation
    - Inside global port number – new source port number after translation
    - Outside global IP address – public destination IP address
    - Outside global port number – destination port number
  - PAT works only with TCP/UDP because of the port numbering
  - ICMP with NAT requires some proprietary methods

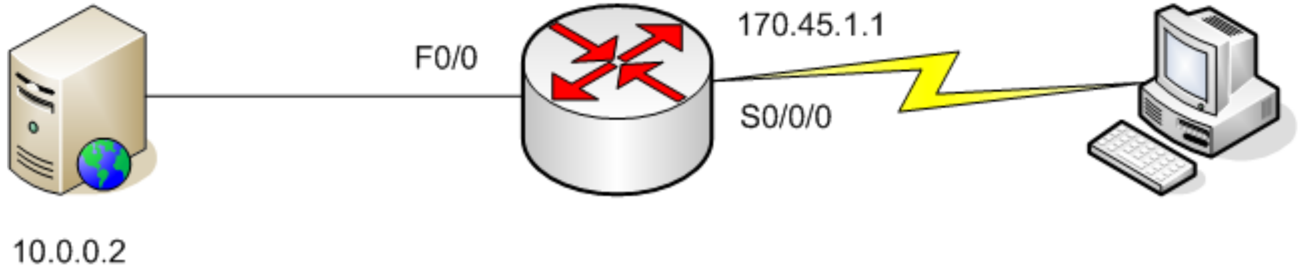
NAT (8/13)

- With the available port addresses, you can have up to 16000 simultaneous connections with PAT
- You should add an ACL to your gateway router, that blocks all the outside traffic that has been spoofed with private IP addresses
- Typically, you use PAT for the traffic that originates from the inside and static NAT for the traffic that originates from the outside
- You can also apply something called Port Address Redirection
  - If you have i.e. a Web Server inside (say 10.0.1.2), which should be accessible from the outside
  - You still have only one public IP address (say 192.200.100.100), but the gateway router must now perform port address redirection
  - A static PAT entry is required in your gateway router
  - An incoming packet destined for 192.200.100.100 with the port 80 is redirected to an inside address 10.0.1.2 with the port 80

NAT (9/13)

- Disadvantages of NAT:
  - Increased complexity
  - Some applications require a real IP address
  - Compatibility with certain applications
  - Problems with security protocols, such as IPSec
  - Peer2Peer appications are harder to setup
  - Performance reduction

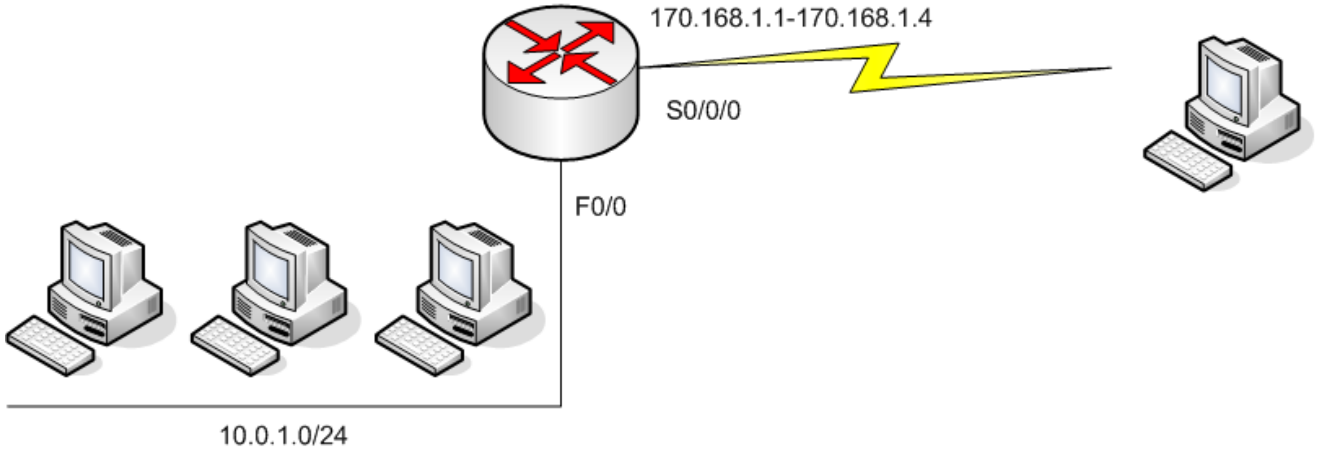
NAT (10/13)



```
Router(config)#ip nat inside source static 10.0.0.2 170.45.1.1
Router(config)#int f0/0
Router(config-if)#ip nat inside
Router(config-if)#exit
Router(config)#int s0/0/0
Router(config-if)#ip nat outside
Router(config-if)#exit
```

Static NAT
The inside Web Server 10.0.0.2 is statically mapped to a public IP address: 170.45.1.1

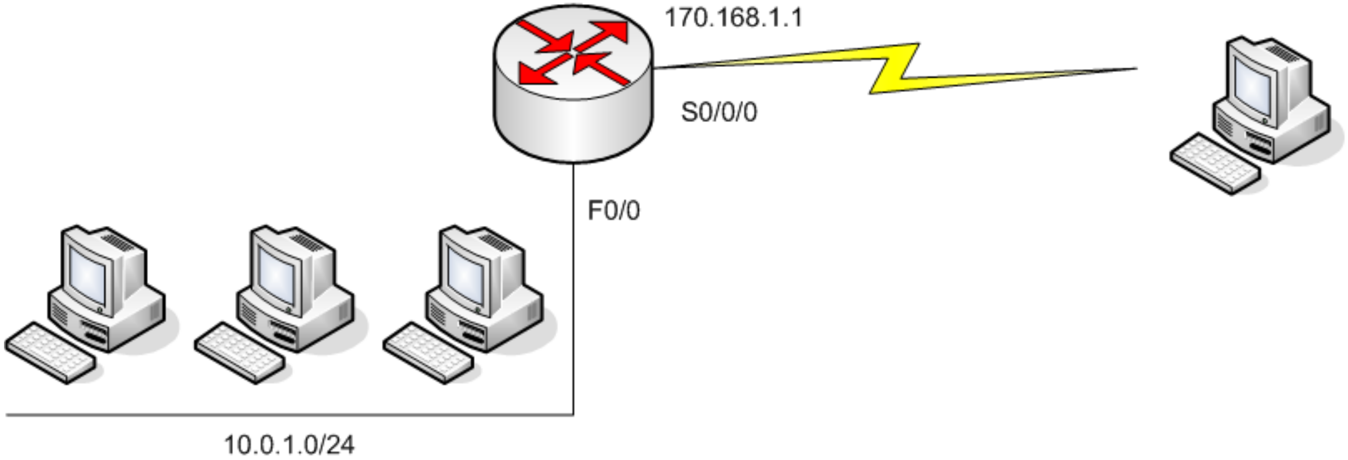
NAT (11/13)



```
Router(config)#ip nat pool cisco1 170.168.1.1 170.168.1.4 netmask 255.255.255.0
Router(config)#ip nat inside source list 1 pool cisco1
Router(config)#int f0/0
Router(config-if)#ip nat inside
Router(config-if)#exit
Router(config)#int s0/0/0
Router(config-if)#ip nat outside
Router(config-if)#exit
Router(config)#access-list 1 permit 10.0.1.0 0.0.0.255
```

Dynamic NAT
A pool with a name (cisco1) is defined with IP addresses and a netmask.
The ”ip nat inside source list 1 pool cisco1” command tells the NAT box (router) to translate IP addresses that match access-list 1 to an address found in the pool named cisco1.
“access-list 1 permit 10.0.1.0 0.0.0.255” creates an access list number 1 which tags an interesting traffic (10.0.1.0/24)

NAT (12/13)



```
Router(config)#ip nat pool cisco2 170.168.1.1 170.168.1.1 netmask 255.255.255.0
Router(config)#ip nat inside source list 1 pool cisco2 overload
Router(config)#int f0/0
Router(config-if)#ip nat inside
Router(config-if)#exit
Router(config)#int s0/0/0
Router(config-if)#ip nat outside
Router(config-if)#exit
Router(config)#access-list 1 permit 10.0.1.0 0.0.0.255
```

PAT

The only difference between PAT and dynamic NAT configuration is that the pool of public IP addresses contains only one address. We have also added a command ”overload” in the end of ”ip nat inside source list…” command

NAT (13/13)

- To debug NAT:
  - – Router#**sh ip nat translation**
- To clear a NAT entry:
  - – Router#**clear ip nat translation ?**
- To clear all NAT entries:
  - – Router#**clear ip nat translation ***
- To negate a NAT command use **no** in the beginning of the command, as usual

DHCP (1/3)

- DHCP info that the server sends the client:
  - IP address
  - Subnet mask
  - Default gateway address
  - DNS domain name
  - 1 or 2 DNS server addresses
  - 1 or 2 WINS server addresses (Windows specific info)
  - Lease length of the IP address

DHCP (2/3)

- Router(config)#**[no] service dhcp**
- Router(config)#**ip dhcp pool pool_name**
- Router(config-dhcp)#**network network_number [subnet_mask| prefix_length]**
- Router(config-dhcp)#**domain-name domain_name**
- Router(config-dhcp)#**dns-server IP_address [#2_IP_address]**
- Router(config-dhcp)#**netbios-name-server IP_address [#2_IP_address]**
- Router(config-dhcp)#**netbios-node-type node_type**
- Router(config-dchp)#**default-router IP_address**
- Router(config-dhcp)#**lease days [hours] [minutes] | infinite**
- Router(config-dhcp)#**import all**
- Router(config-dhcp)#**exit**
- Router(config)#**ip dhcp ping timeout time_ms**
- Router(config)#**ip dhcp excluded-address first_IP_address last_IP_address**

DHCP (3/3)

- You must have configured a router interface (IP address) and enabled it before you can apply a DHCP server
- IOS can figure out from the IP settings of an interface, where to apply a DHCP pool

Preassignment

- Preassignment
  - T-110_5101_preliminary_2.xlsx (MS-Excel)
  - Subnetting exercise. Given a larger subnet block, subnet the network
  - Apply VLSM to fit in to the given block size
  - The how-to can be found for example in: Todd Lammle: CCNA Study Guide. 6th edition. ISBN: 978-0-470-11008-9. Ch3: Subnetting, VLSM
  - Present your solution in the demo session

Literature

- http://www.cisco.com/warp/cpropub/45/tutorial.htm
- http://www.routersim.com/CCNA6_Supported_Commands.html
- Richard Deal: CCNA Study Guide. 3rd edition. ISBN-13: 978-0071497282
- Todd Lammle: CCNA Study Guide. 6th edition. ISBN: 978-0-470-11008-9.
  - Only ch 3: Subnetting, VLSM
- Scott Empson: CCNA Portable Command Guide. 2nd edition. ISBN-13: 978-1587201936. The copy of this book is in the lab class. You should check your IOS command sequence from this book before invoking the commands
- http://www.learnios.com/index.php