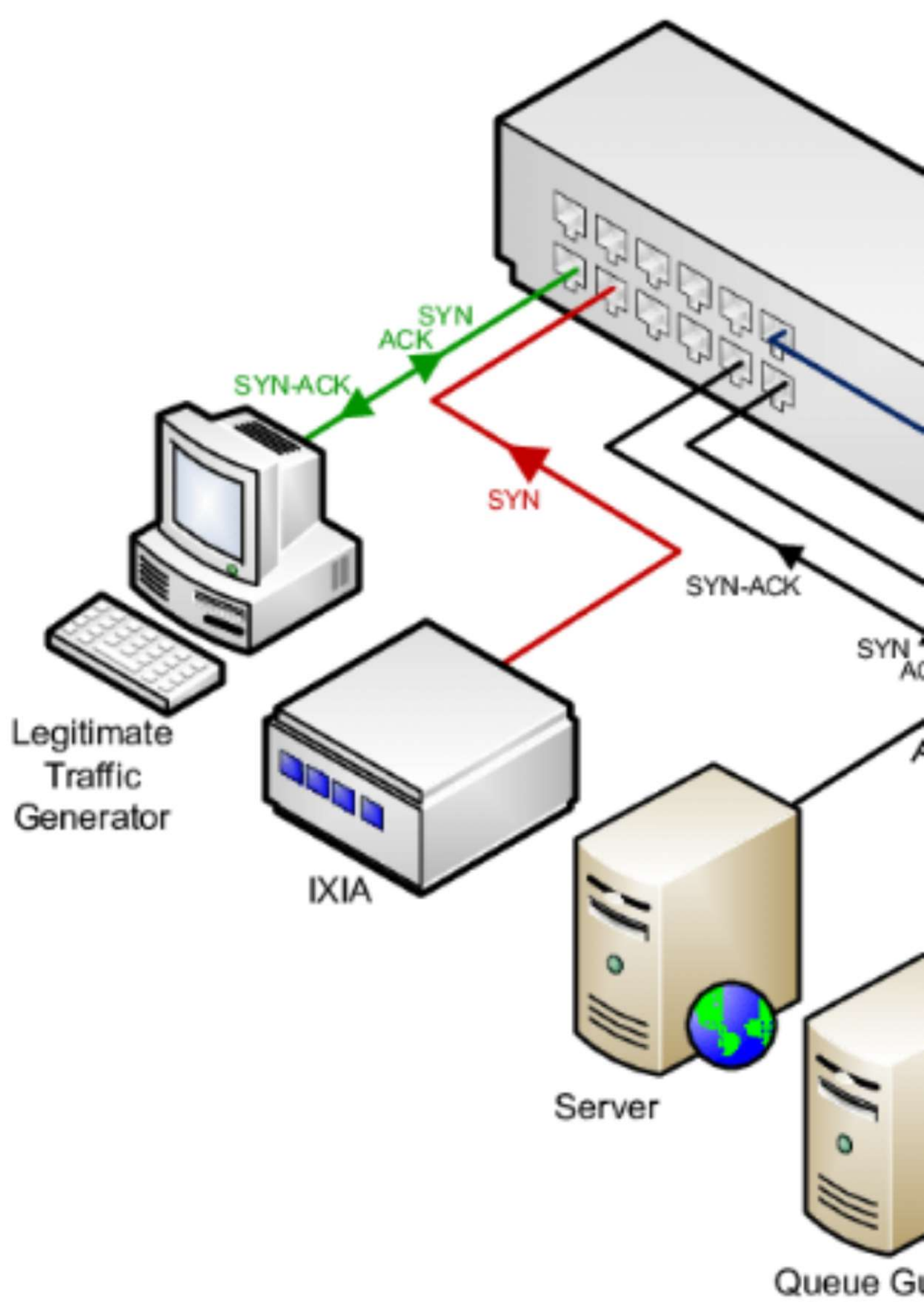

SYN
ACK
SYN-ACK
SYN
SYN-ACK
Legitimate
Traffic
Generator
IXIA
Server

Antonio Cianfrani

Laboratory of Network Design and Configuration

A.Y. 2022/2023

Info

- **Lectures schedule**
  - Friday 14:00 – 19:00 (Classroom 22) → **14:45 – 19:00**

- **Antonio Cianfrani**
  - Room 220 - DIET Department - II floor
  - E-mail: antonio.cianfrani@uniroma1.it
  - Tel.: 0644585371

Tentative scheduling

Date	Topic
February 24, 2023	Networking fundamentals - Cisco Packet Tracer software
March 3, 2023	Router IP Configuration
March 10, 2023	RIP and RIPv2
March 17, 2023	OSPF
March 24, 2023	DHCP and NAT
March 31, 2023	Training lab activity - part I
April 7, 2023	Eastern Holydays
April 14, 2023	Midterm exam part I
April 21, 2023	Ehternet, Management and Virtual LAN (VLAN)
April 28, 2023	Access Control List (ACL)
May 5, 2023	Access Control List (ACL)
May 12, 2023	Security
May 19, 2023	Training lab activity - part II
May 26, 2023	Midterm exam part II

Objectives

- The main aim of the LNDC course is to provide practical approach for the management of an Internet Sercive Provider (ISP) network and of a LAN network.
- More in details:
  - to configure an IP network from addressing to routing and troubleshooting
  - to provide advanced configurations of an Ethernet network from Virtual LAN to security solutions
- The course (6 CFU), is composed of two parts
  - Part I (3 CFU): router IP
  - Part II (3 CFU): switch Ethernet and Security

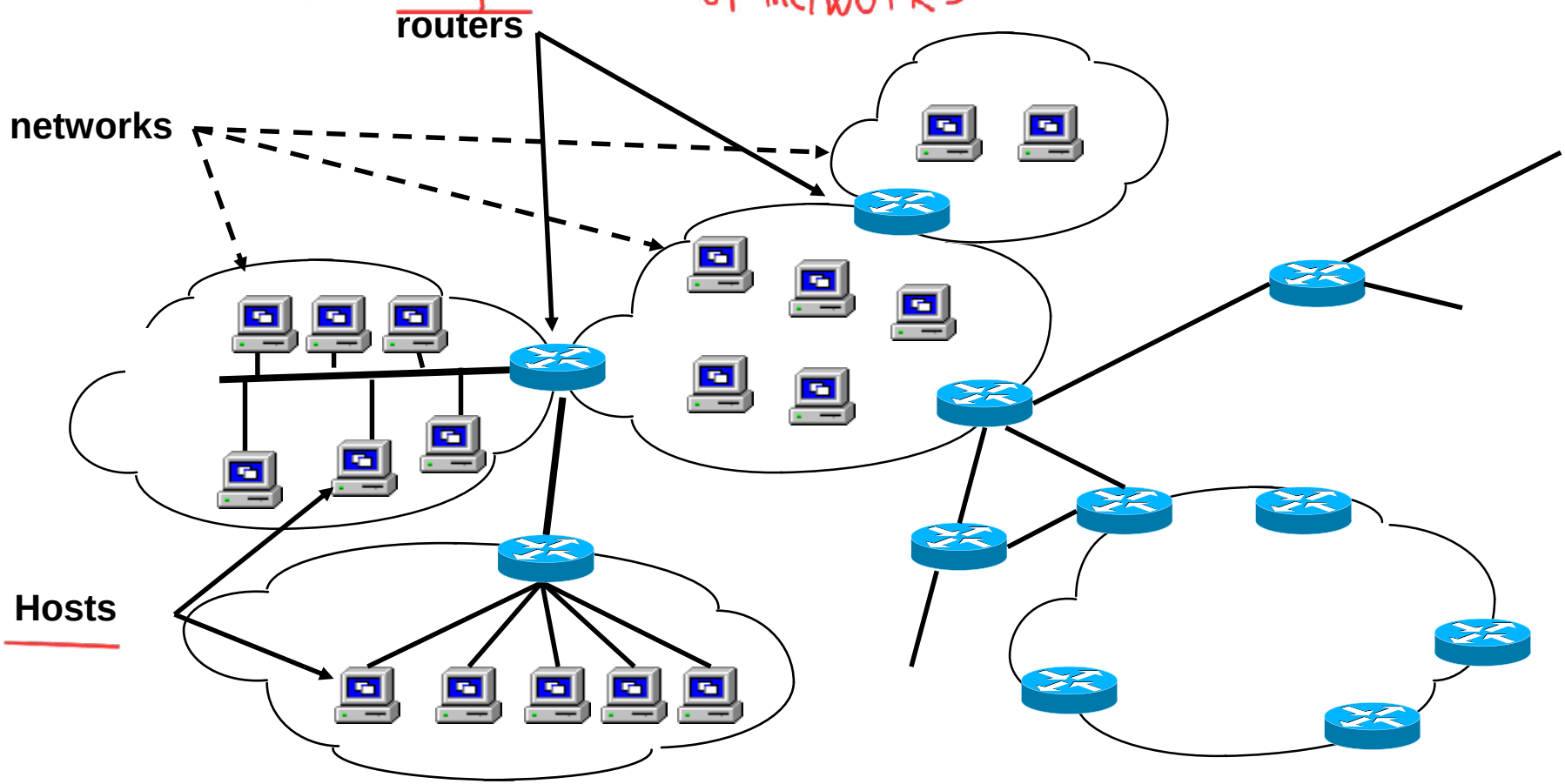
Antonio Cianfrani

Fundamentals of IP Networking

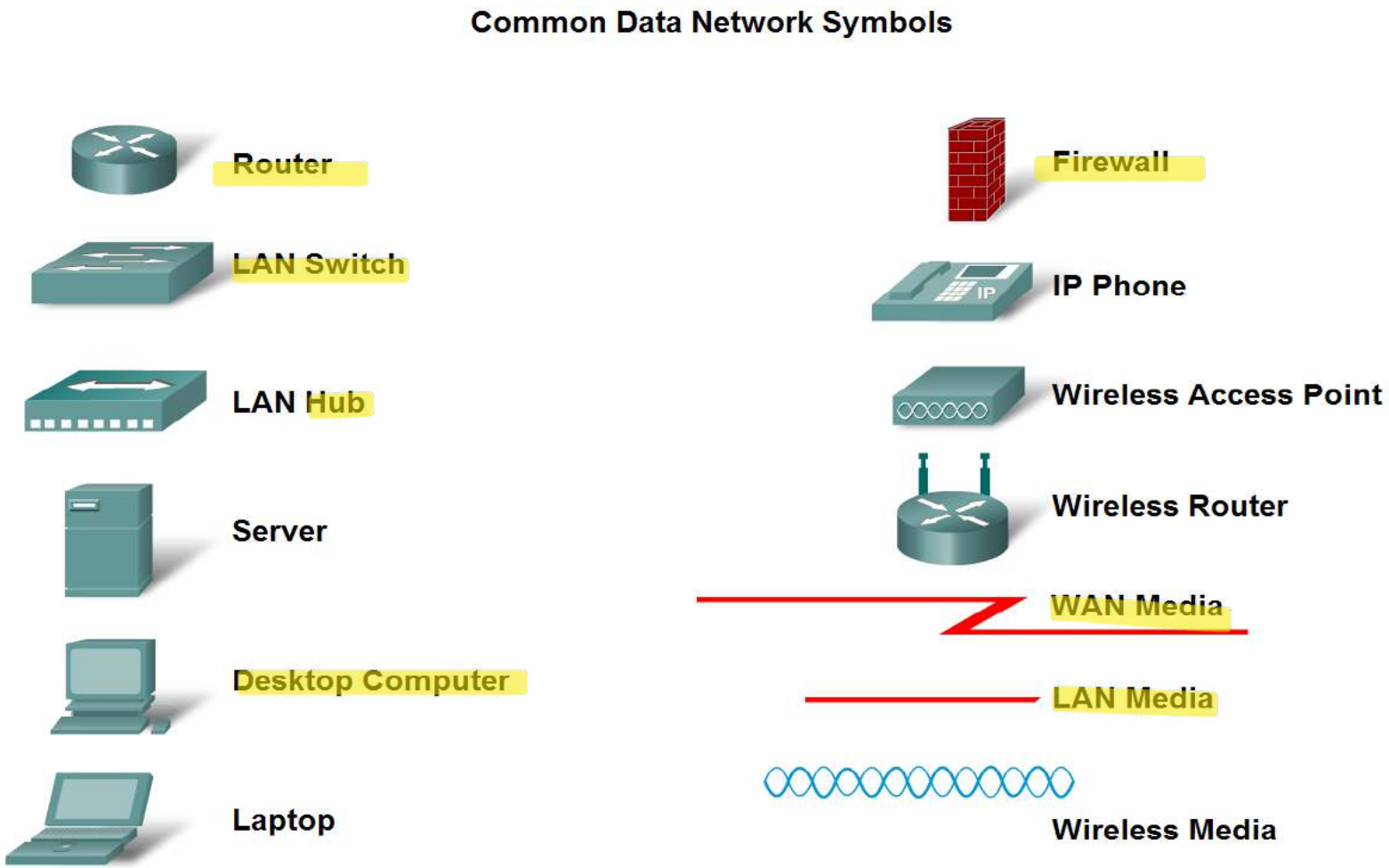
The Internet network

main devices
on internet network

interconnection
of networks



Network devices

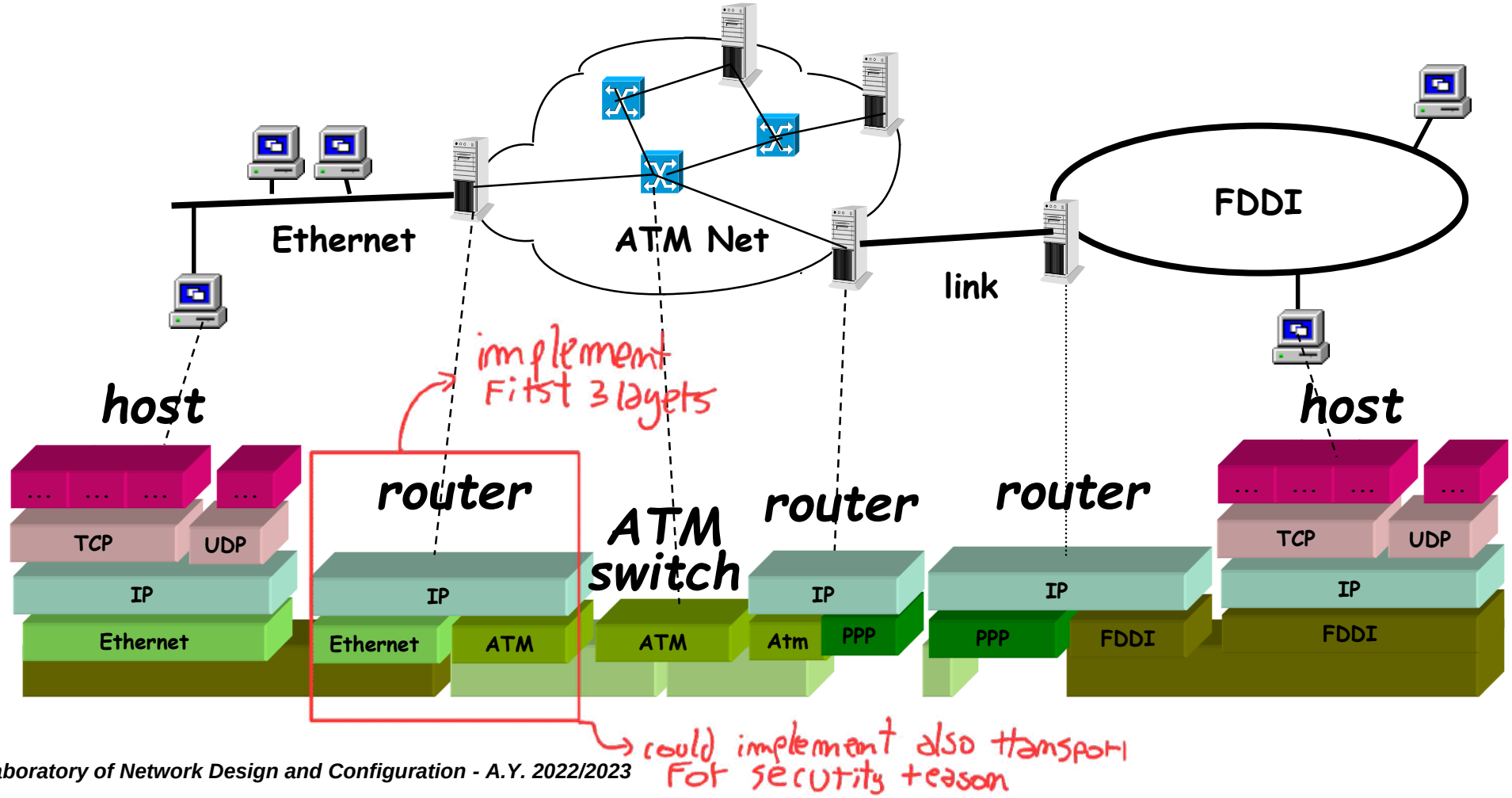


TCP/IP protocol stack

Application only use concept of source and destination port
Transport (TCP/UDP)
Network (IP) IPV4 we use
Data Link ↳ we configure internet switches

- Applications (HTTP, FTP, TELNET): distributed programs using the network to exchange data (messages).
- Transport layer: end-to-end communication service; UDP is connectionless, TCP is connection oriented (take care of segments re-orderding and recovery).
- Network layer main functions: addressing and forwarding/routing.
- Data Link layer: set of protocols for different physical networks/links (Ethernet, PPP,…)

TCP/IP stack in action



main focus

Router functions

- A router makes possible the interconnections among different subnetworks
- A router implements the first 3 layers of the TCP/IP protocol stack. The main functions performed are the ones of **Network layer**.
- Main protocol of Network layer: Internet Protocol (IP).
- The IP layer (RFC 791, 919, 922, 950, 1349) executes the following functions:
  - **Addressing** → IP protocol define IP addresses, follow a hyerarchy structure
  - **Forwarding** → computing routing paths, using also protocols
  - **Fragmentation**
- A router must be properly configured by the network administrator

Addressing (1/3)

- An IP Address identify routers and hosts: carry network device has unique identifiers
  - a host connected to more than a subnetwork (multi-homed) has an IP address for each subnetwork;
  - a router has an IP address for each interface.
- An IP address is unique in the Internet (one exception, NAT)
  - it is 32 bits long;
  - a decimal dotted notation is used for easiness.

Binary notation 10010111 01100100 00001000 00010010 32 bits

Dotted notation 151. 100. 8. 18 → 8 bits each

infocom.uniroma1.it

Addressing (2/3)

- IP addresses have a **hierarchical** structure.

- The addressing space is divided into blocks: a block of addresses is assigned to each subnetwork

IP_Address=Net_Id.Host_Id

  - The Net_Id identifies the subnetwork (block): Prefix
  - The Host_Id identifies the specific address of the block

- It allows for a scalable forwarding management
  - Each subnetwork is represented by its Net_Id prefix

Addressing (3/3)

- To identify the Net_ID lenght, the **Subnet Mask (Mask)** is introduced: a 32 bit string having 1s for Net_Id bits and 0s for Host_Id bits

Address = Net_Id . Host_Id

Mask= 1111….11 . 00….000 Ex. /30

The address used to identify the whole block (subnet)

Block Address(S)= Net_Id . 0….000

Mask= 11……..11 . 0….000

- Given the IP address of an host (Add(H)) and a subnet S represented by (Add(S),Mask(S)), to verify if H belongs to S (Matching):

$$Add(H) \in S \Leftrightarrow Add(H) \,\&\, Mask(S) = Add(S)$$

AND

Classfull addressing

↳ DOESN'T NEED NETMASK, ONLY LOOK AT 1ST BYTE

IP Address Classes

Address Class	1st octet range (decimal)	1st octet bits (green bits do not change)	Network(N) and Host(H) parts of address	Default subnet mask (decimal and binary)	Number of possible networks and hosts per network	
A	1-127** FIRST BYTE OF IP ADDR.	00000000-01111111	N.H.H.H = 8	255.0.0.0	128 nets (2^7) 16,777,214 hosts per net (2^24-2)	/8
B	128-191	10000000-10111111	N.N.H.H = 16	255.255.0.0	16,384 nets (2^14) 65,534 hosts per net (2^16-2)	/16
C	192-223	11000000-11011111	N.N.N.H = 24	255.255.255.0	2,097,150 nets (2^21) 254 hosts per net (2^8-2)	/24
D	224-239	11100000-11101111	NA (multicast)			⋮
E	240-255	11110000-11111111	NA (experimental)			

** All zeros (0) and all ones (1) are invalid hosts addresses.

Classless and subnetting

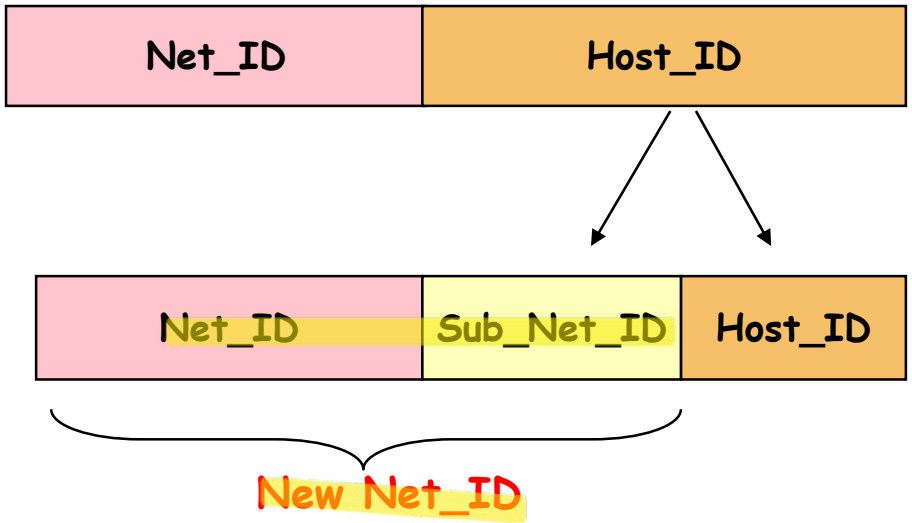
- Subnetting allows for the use of netmask different than classfull ones:

  Esempio.: subnetting starting from a class C block:

  255.255.255.128 (10000000) ⟷ /25

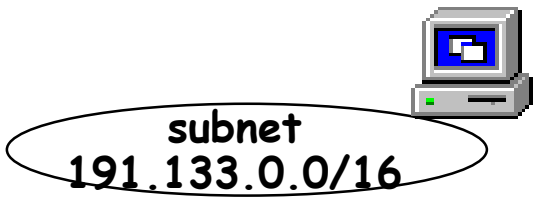
  255.255.255.192 (11000000) ⟷ /26

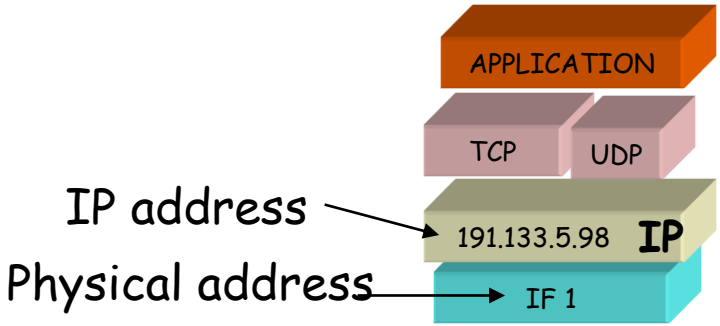
  ……



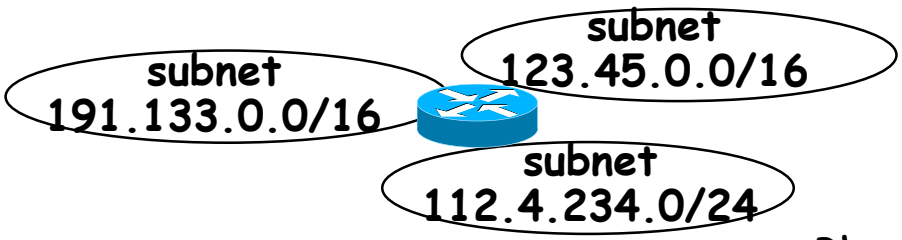
Host vs router

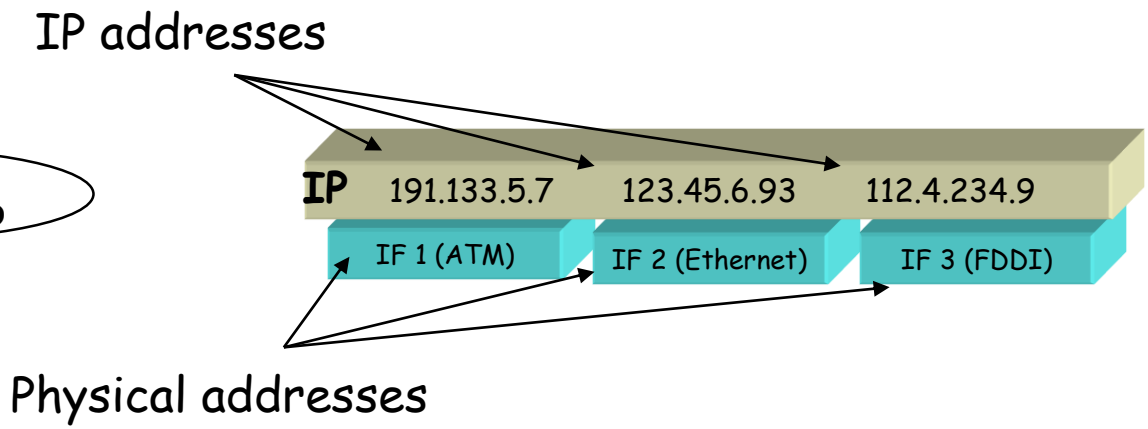
- **host**





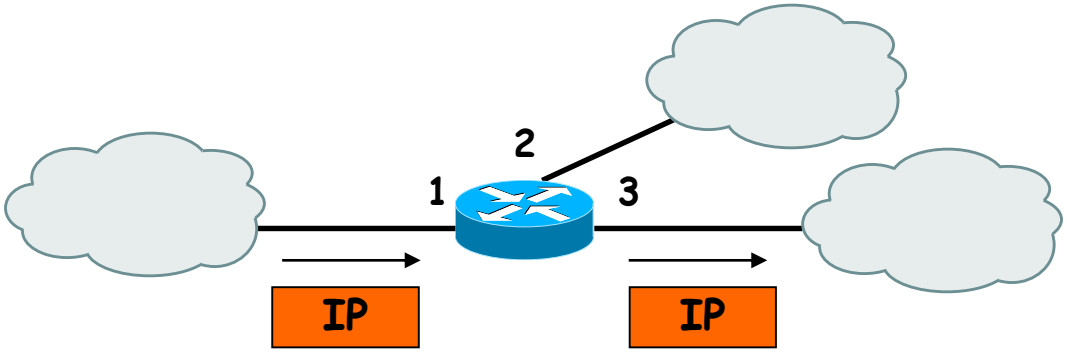
- **Router**





IP Forwarding and routing

- The IP routers perform packet forwarding:

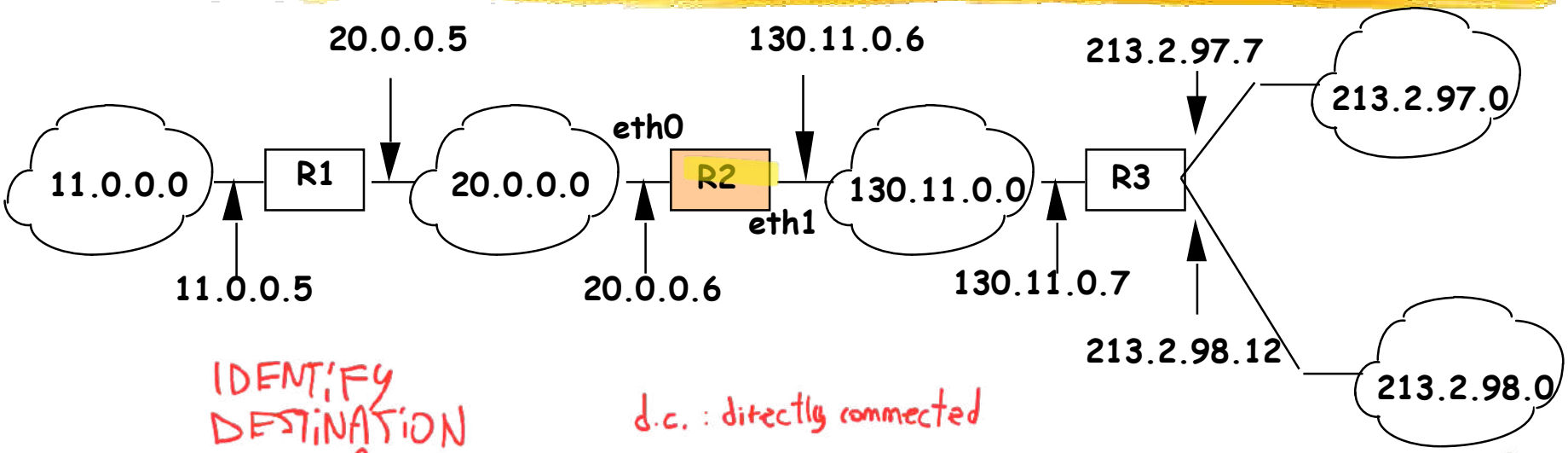


- Routing: the path form a source host to a destination host is composed of routers and networks crossed.

Routing table

- IP routing is based on the routing table (each host and router has its own one): routing information about known destinations
- A routing table is composed of (N, M, NH, I) entries:
  - N is the IP address of the destination network,
  - M is the subnet mask of the destination network having N as address
  - NH is the next-hop router toward the destination
  - I is the router/host interface to which forward the packets directed to network N
- **Longest Prefix Matching rule**

Example



IDENTIFY DESTINATION

d.c. : directly connected

R2 routing table

ROUTING FOR DESTINATION

N	M	NH	I
20.0.0.0	255.0.0.0	d.c. (20.0.0.6)	eth0 → local interface
130.11.0.0	255.255.0.0	d.c. (130.11.0.6)	eth1
11.0.0.0	255.0.0.0	20.0.0.5	eth0
213.2.97.0	255.255.255.0	130.11.0.7	eth1
213.2.98.0	255.255.255.0	130.11.0.7	eth1

Default router/gateway

MATCHING, (!= NOT)

D = DEST IP ADDRESS ⟶ ROUTER DO A MATCHING FOR ROUTING D AND M(1) = N(1)

- Router to use as next hop for traffic directed to an unknown network (not present in the routing table)
- Not mandatory but often used.

↳ WE TAKE LONGEST PREFIX MATCHING

- It is representd by all 0s for columns N and M.

WITHOUT DROP IF NO MATCHING

Destinazione N	Maschera M	Next hop NH	Interfaccia I
20.0.0.0	255.0.0.0	d.c. (20.0.0.6)	eth0
130.11.0.0	255.255.0.0	d.c. (130.11.0.6)	eth1
0.0.0.0	0.0.0.0	20.0.0.5	eth0

↳ INTERNET CONNECTION

↳ D AND 0.0.0.0. = 0.0.0.0. ⇝ MATCH OF ZERO BITS, NETMASK IS 0!
ALWAYS !!!

↳ SHORTEST MATCHING

Routing table of a Cisco router

SOURCE OF THIS INFORMATION IS GIVEN WHEN ASSIGN IP ADDRESS

Directly Connected Route

Dynamically Updated Route

Static Route

Default Route

Edit Router C

Physical | Config | CLI

IOS Command Line Interface

```
Router C#show ip route
Codes: C - connected, S - static, I - IGRP, I - RIP. M - mobile, B
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter a
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external typ
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - E
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is 172.8.1.1 to network 0.0.0.0

     172.8.0.0/24 is subnetted, 1 subnets
C       172.5.1.0 is directly connected, Serial2/0
     172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
R       172.16.0.0/16 [120/1] via 200.1.1.1, 00:00:27, Serial3/0
S (STATIC)  172.16.1.0/24 [1/0] via 200.1.1.1 NEXTHOP
     172.64.0.0/24 is subnetted, 1 subnets
C       172.64.1.0 is directly connected, FastEthernet0/0
C    200.1.1.0/24 is directly connected, Serial3/0
S*   0.0.0.0/0 [1/0] via 172.8.1.1
Router C#
```

Copy | Paste

Host configuration

- Any host connected to an IP network must be properly configured
  - Host name
  - IP address
  - Subnet mask
  - Default router
  - Server DNS
- DHCP protocol → ASSIGN IP ADDRESS DYNAMICALLY, AND ALSO THE OTHER INFORMATION
- Private IP addresses: NAT (NETWORK ADDRESS TRASLATION)

  → ARE PRIVATE IP OF AN HOST

IP routers

Router: overview

- A router is very similar to a PC from an hardware point of view: CPU, memories, system bus and input/output interfaces.
- It has a dedicate operating System: IOS (Internetwork Operating System) for the Cisco devices.
- The IOS must be configured to properly work: IP addresses, routing protocols setting, security aspects…..
  - Configuration files

ROUTER MUST BE CONFIGURED, SWITCH MAY BE NOT

Router hardware components (1/4)

- **CPU**
  - Execute IOS instructions, such as initialization, routing functions and interfaces management.
  - High performance routers can have more than one CPU.
- **RAM** (RANDOM ACCESS MEMORY)
  - Used to store the routing table, the cache fast switching, the configuration file and the packet queues.
  - It provides the run time space for the software executed by the IOS and it is composed of interfaces shared memory and main processor memory.
  - It stores the configuration file of the router while it is powered on: the running configuration file.

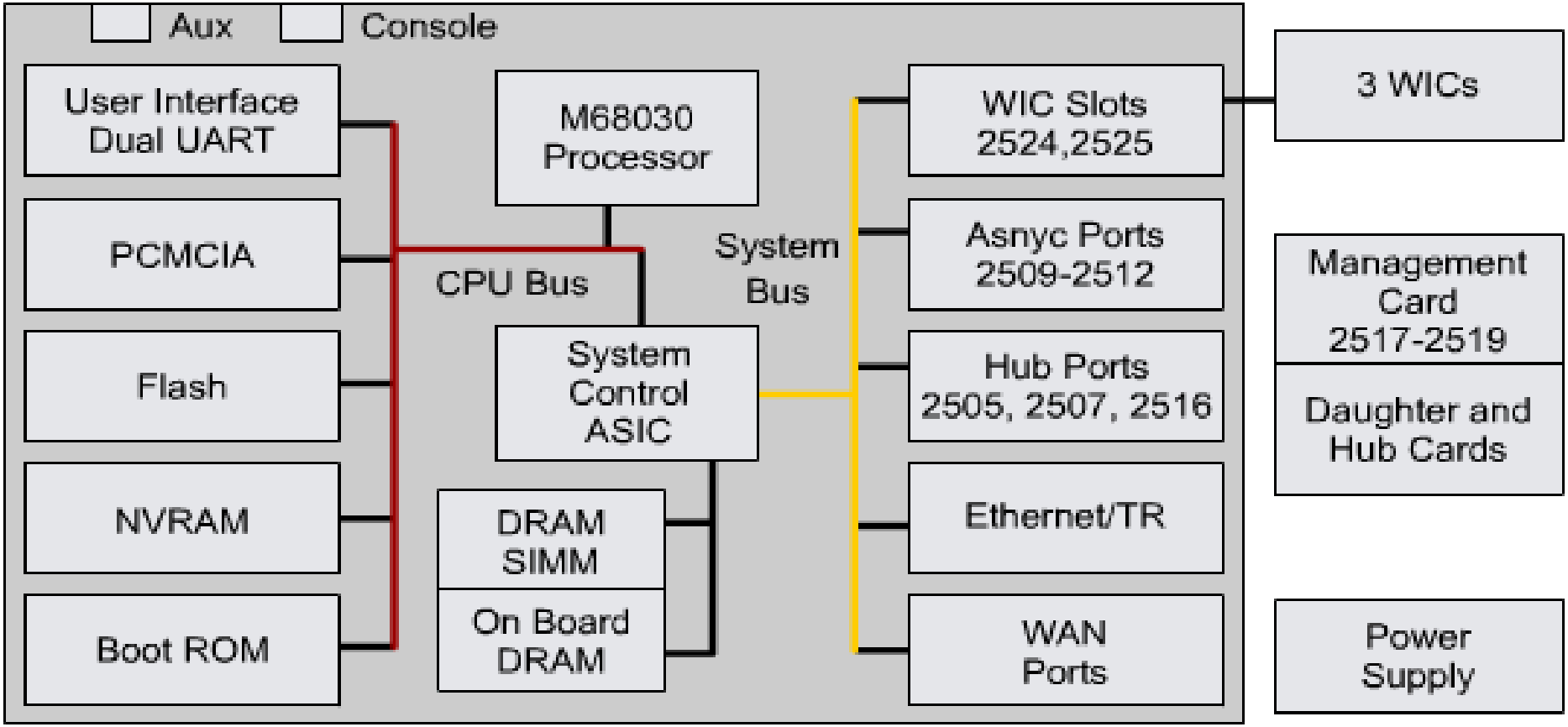
Router hardware components (2/4)

- **Flash**
  - Used to store the image of the IOS (the image will be loaded every time the router is powered on)
- **NVRAM** (NON VOLATIVE RAM)
  - It store the configuration to be used at startup: the <u>startup configuration file</u>.
- **Bus**
  - Usually two buses: the system bus and the CPU bus.
  - The system bus is the one providing communication among CPU and interfaces and used to forward packets among ingress and egress interfaces.

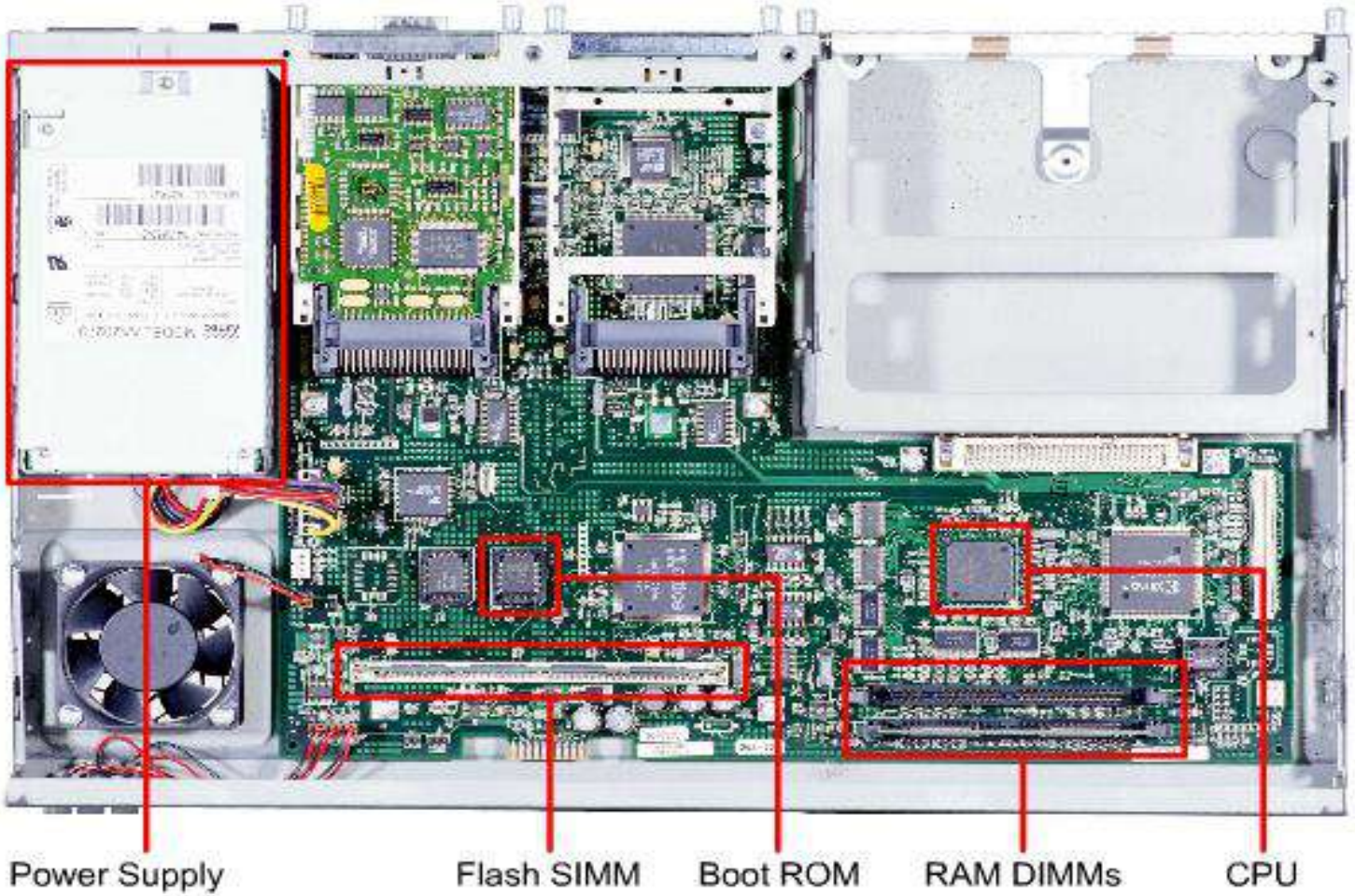
Router hardware components (3/4)

- **ROM**
  - Stores in a permanent way the code used to perform hardware integrity checking (performed at startup)
- **Interfaces**
  - the connections among the router and the "external world"
  - different types of interfaces
  - Can be classified in: LAN, WAN and console/AUX

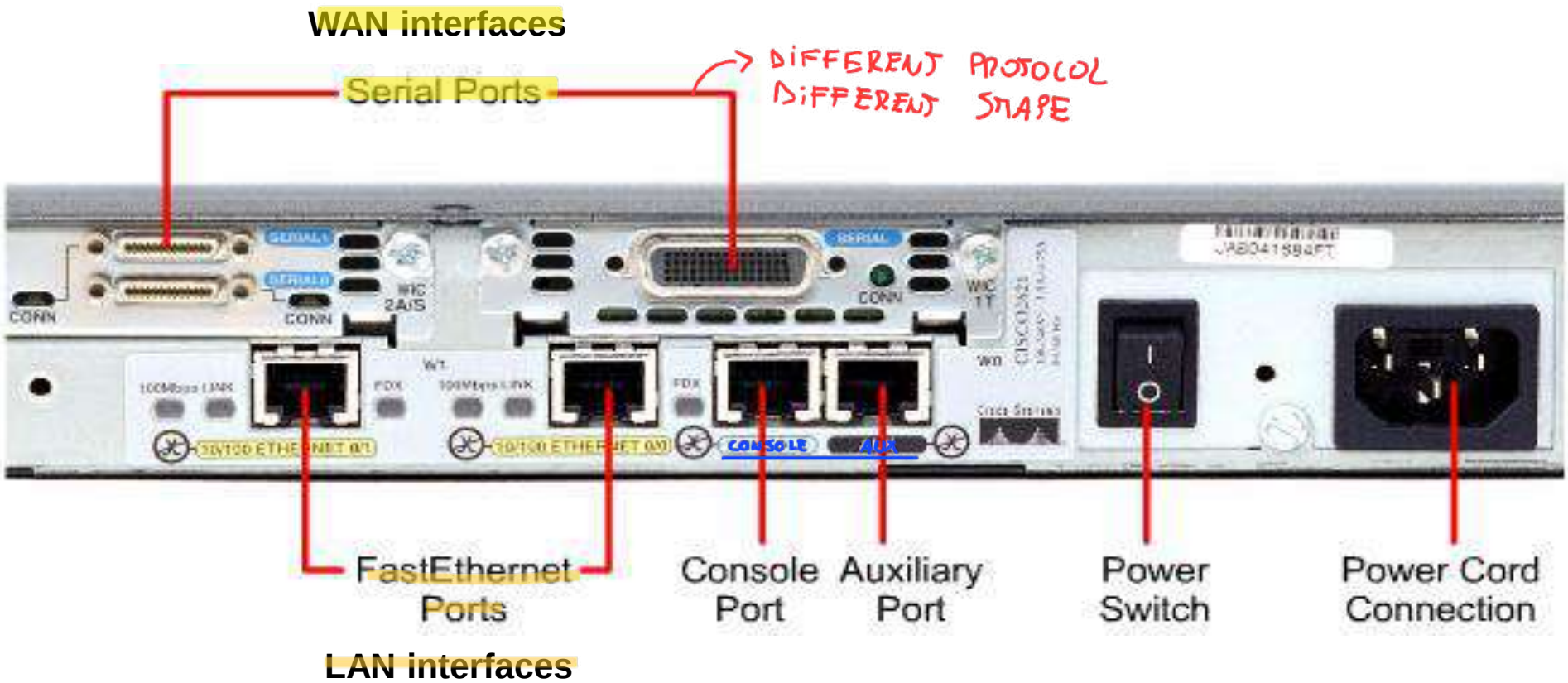
Router hardware components (4/4)



Cisco 2600 router

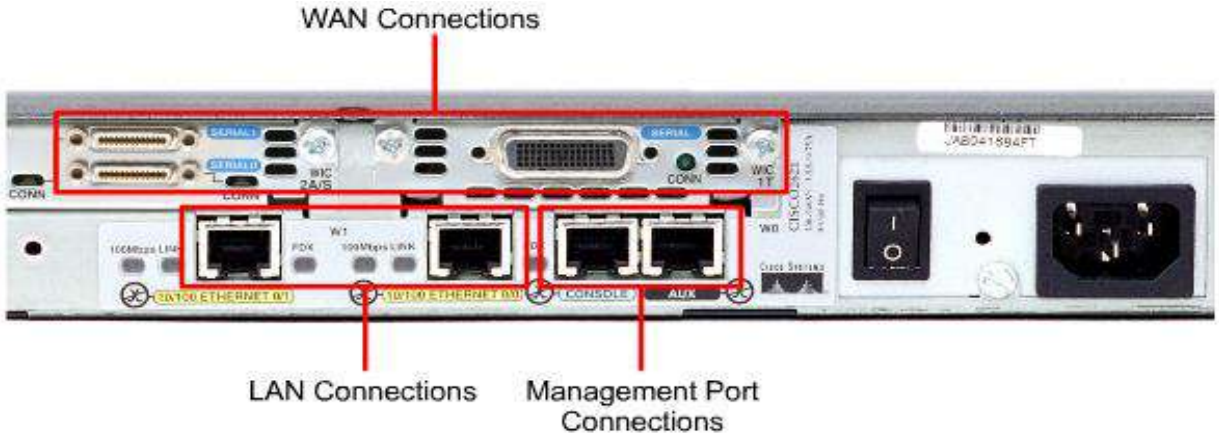


Cisco 2600 router interfaces



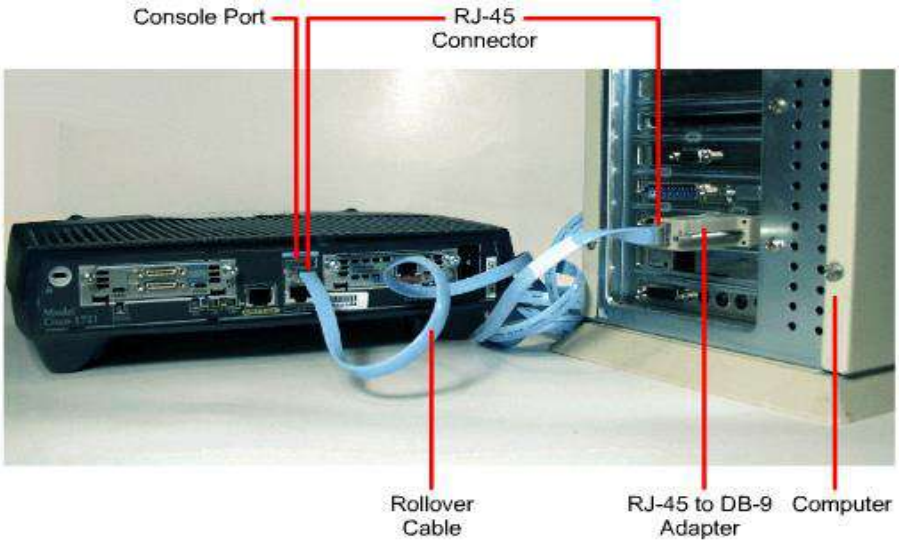
Router interfaces

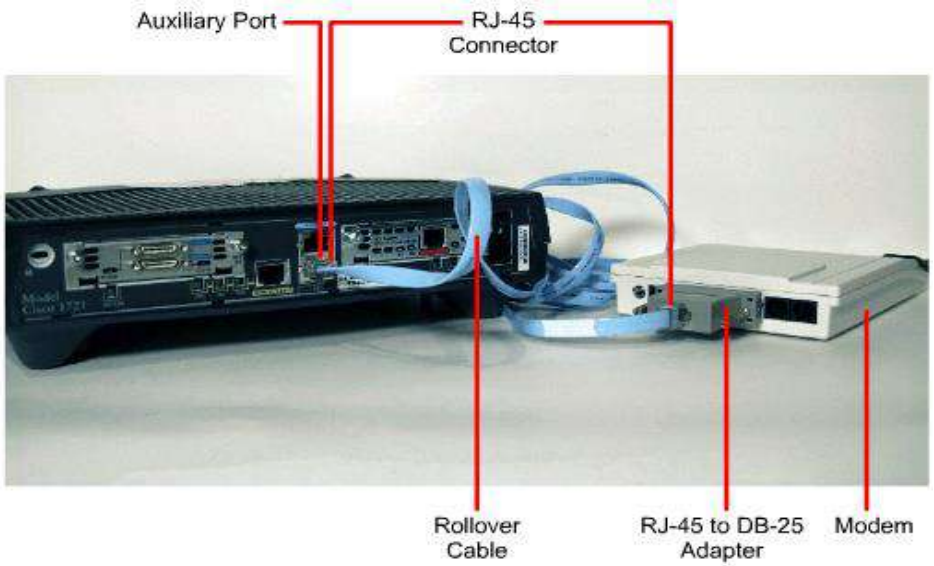
- **LAN:** provide connectivity toward a Local Area Network (LAN); the de-facto LAN standard is Ethernet
- **WAN:** provide connectivity toward a Wide Area Network (WAN); there is no de-facto standard (many different protocols); this is usually the connection toward an Internet Service Provider.
- **management:** provide text-based connection toward a PC to configure the router



Management interfaces

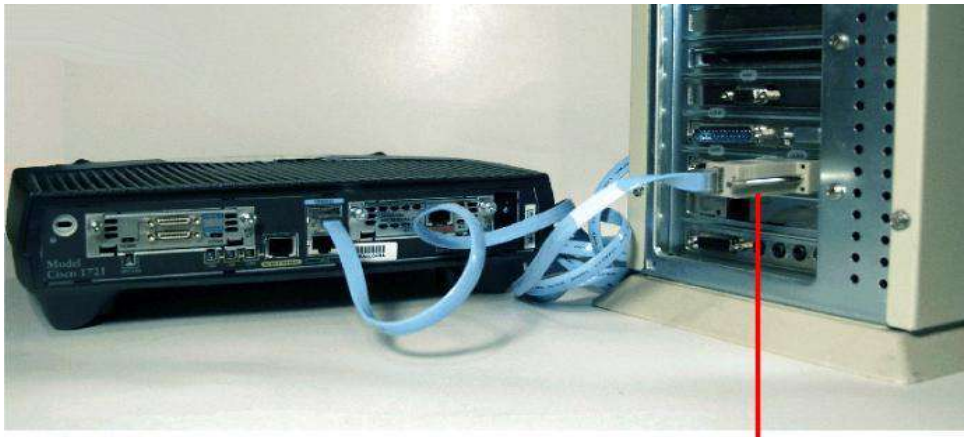
- The console and AUX (auxiliary) interface are dedicated to router management operations.
- There must be already one (the console) for the initial router configuration.



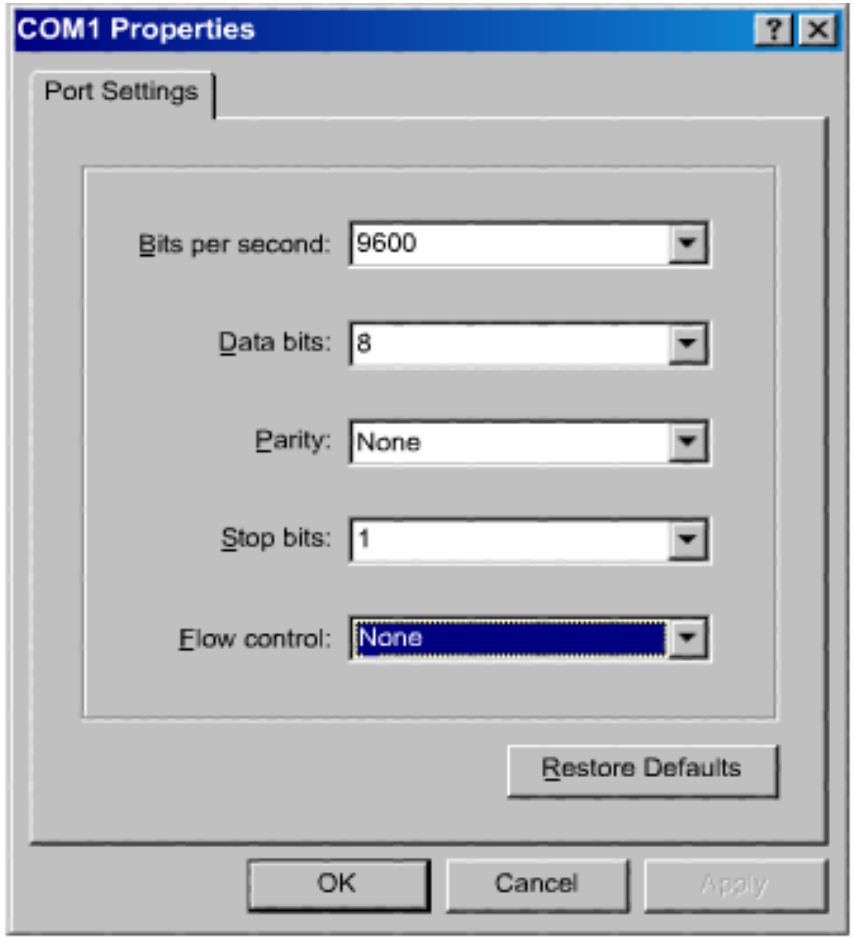


Console interface

- The access provided by the console interface is defined as out-of-band
- It is used for the initial configuration, for troubleshooting, for monitoring and for disaster recovery procedures.
- A rollover cable and a "RJ-45 to DB-9" adapter are needed to connect to a console port.
- The PC must support the terminal emulation (WINDOWS Hyperterminal).

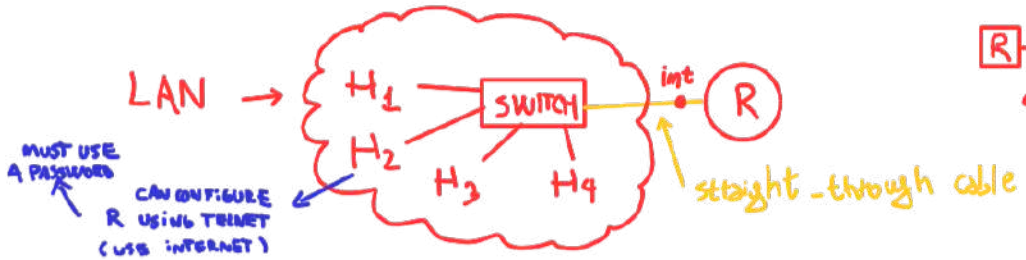


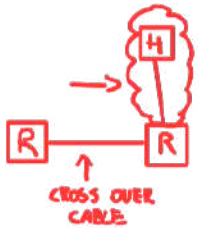
Hyperterminal interface



The LAN interface

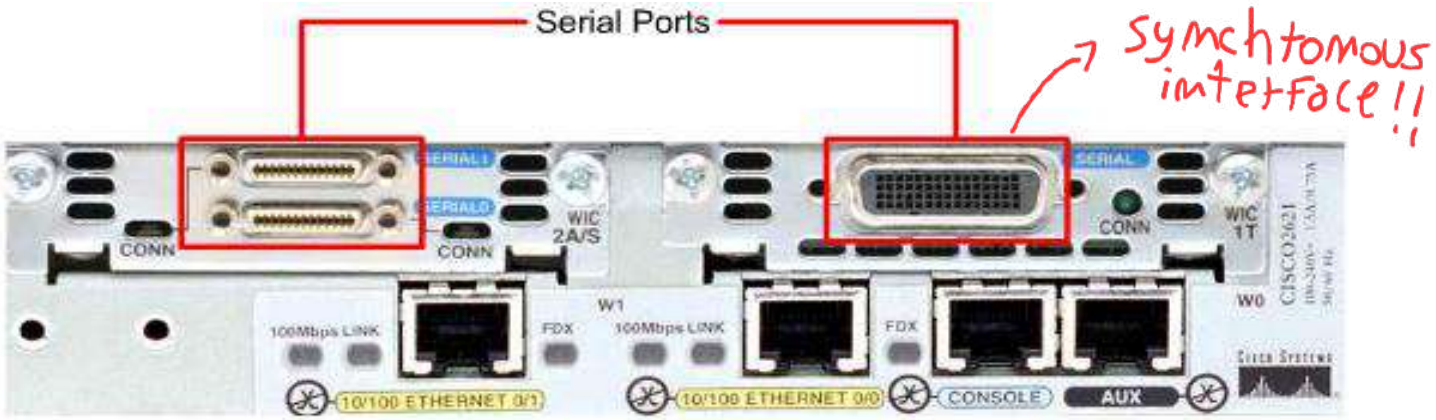
- The router is connected to a LAN using the Ethernet (FastEthernet) interface.
- The router is connected by its LAN interface to a switch, a hub or a host/router (very rare).
- For connections with a switch or hub, a straight – through cable is used.
- For connections with a router or a host, a crossover cable is used
- The LAN interface allows the in-band router configuration: TELNET



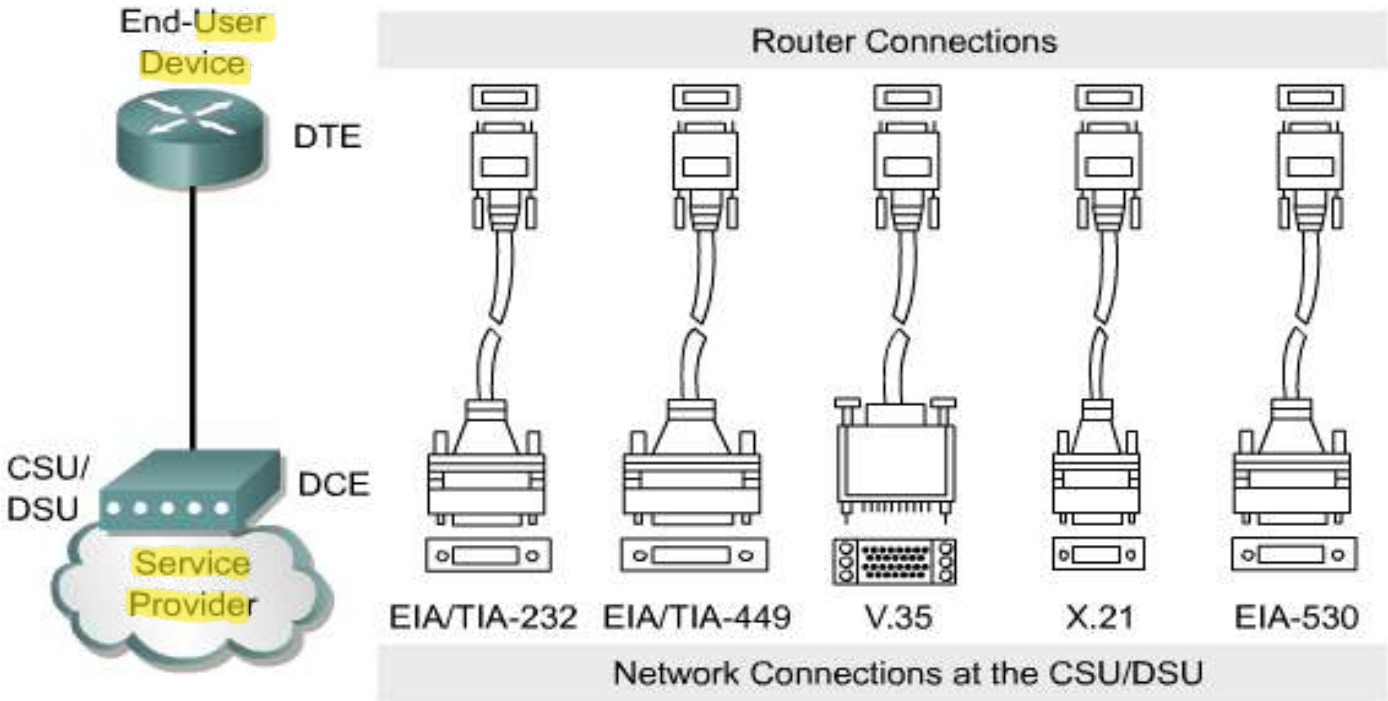


The WAN interface (1/2)

- The WAN connections (also defined as "Serial Ports") can be of different types since different technologies (protocols) can be used.



The WAN interface (2/2)



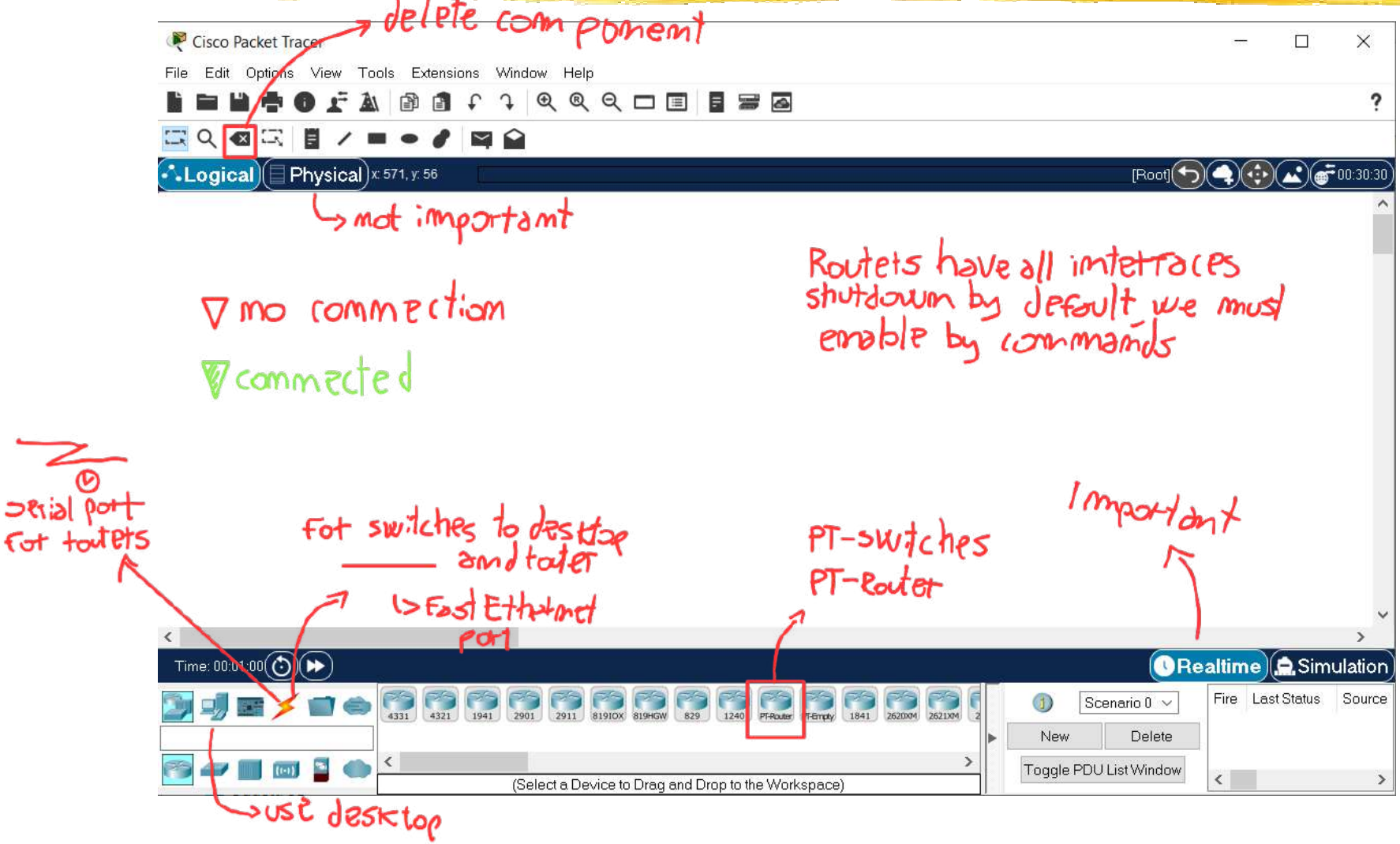
Antonio Cianfrani

Packet Tracer and initial router configuration

Packet Tracer (1/2)

- **Packet Tracer?**
  - Cisco Packet Tracer is a software able to emulate CISCO networking devices.
- **Packet Tracer features:**
  - Allows to create network topologies with Cisco devices and generic device.
  - Emulate the Command Line Interface (CLI) of the Cisco IOS (a subset of functions).
  - Allows to configure, using the CLI (or the GUI), the network devices so that to check the correctness of a network configuration .
  - Dynamically evaluate the device state and the traffic packets exchanged into the network.

Packet Tracer (2/2)



Topology and available devices

- **It is possible**
  - To start from an existing topology
    - open -> Reference_Topologies
  - Or to create your own network
- **It is possible to use**
  - Cisco network devices with specific hardware features
  - Customize generic devices

Connecting devices

- To connect two network devices, it is needed to select:
  - An adequate physical medium (wired/wireless cable)
  - The proper interfaces
- It is also possible to use the smart connection mode:
  - Packet Tracer selects on itself the cable and the interfaces to be connected

Configuring a device

- Packet Tracer provides a set of basic configuration commands to be performed by means of the graphical interface (GUI- Grafic User Interface).
- On the bottom, the list of commands to be used for the CLI is shown.

Configuring a Cisco Router

CLI (1/2)

- To configure a router/switch we need to learn how the Command Line Interface (CLI) is organized
- How to reach the CLI?
  - Console Session
  - Telnet Session

constraint on be on same place of router

can be done remotly

CLI (2/3)

- **The CLI has a hierarchical structure**
  - Different modes with different configuration levels
    - User EXEC mode ⇝ only see some detail
    - Privileged EXEC mode or enable mode ⇝ for start configuration
    - Global Configuration mode ↳ highest layer

CLI (3/3)

- **The User EXEC only allows for basic monitoring functions ("view only" mode)**
  - It is not possible to perform configuration actions
  - It is identified by the prompt ">".
- **The Privileged EXEC mode allows for more operational functions**
  - It is possible to secure with a password
  - It is identified by the prompt "#"
- **To move form User mode to Privileged mode**
  - "enable" command

Moving among modes

- To come back to Privileged mode from global configuration: "exit" or "Ctrl-Z"

Router

```
Router con0 is now available.

Press RETURN to get started.

User Access Verification
Password:
Router>                    ← User-Mode Prompt
Router>enable
Password:
Router#                    ← Privileged-Mode Prompt
Router#disable
Router>
Router>exit
```

Configuration mode

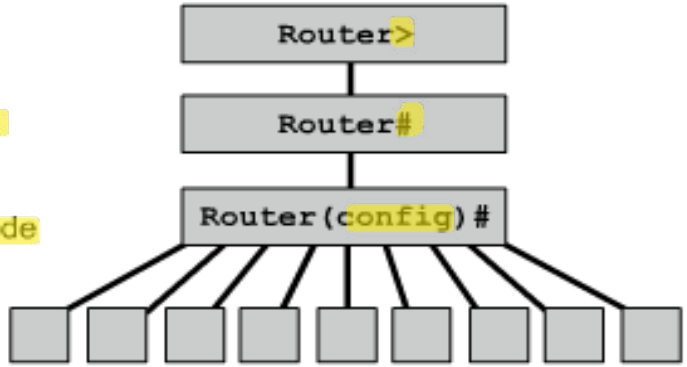
- The global configuration mode allows to perform configuration commands
- To move from the Privileged mode to the Global Configuration one:

```
Router#configure terminal
Router(config)#
```

- From the Global Configuration mode it is possible to configure: interfaces, routing protocols, etc..

Summary

- User EXEC mode
- Privileged EXEC mode
- Global configuration mode
- Specific configuration modes



Configuration Mode	Prompt
Interface	`Router(config-if)#`
Subinterface	`Router(config-subif)#`
Controller	`Router(config-controller)#`
Map-list	`Router(config-map-list)#`
Map-class	`Router(config-map-class)#`
Line	`Router(config-line)#`
Router	`Router(config-router)#`
IPX-router	`Router(config-ipx-router)#`
Route-map	`Router(config-route-map)#`

CLI Help

- The Question Mark (?) is used to list the set of available commands
  - After "—More—" other commands
  - "Enter" to see line by line
  - "Space" to see the whole next page
- It is possible to cut off a word ("ena" instead of enable)
- "Ctrl-P" o "Up Arrow" to recall the last commands
- The ^ symbol show the wrong command part
- "***no ip domain-lookup***" to avoid CLI stops when a wrong command is typed.

Router name

- Each route has a nome (Router is the default name)

- TO assign a different name:
  - From Global configuration mode

```
Router(config)#hostname Tokyo
Tokyo(config)#
```

Securing router access (1/4)

- It is possible to configure a password for router access
- The password can be defined for different access ways:
  1. Console port
  2. Telnet access, referred to as Virtual Terminal Line (vty) in the CLI
  3. Privileged EXEC mode

Securing router access (2/4)

- Password for the console port:

```
Router(config)#line console 0
Router(config-line)#password <password>
Router(config-line)#login
```

- Password for the telnet access (virtual terminal line):

```
Router(config)#line vty 0 4
Router(config-line)#password <password>
Router(config-line)#login
```

UP TO 5 CONTEMPORARY TELNET SESSIONS 0...4

Securing router access (3/4)

- Two possibilities to configure a password for the Privileged mode:

1. "**enable password**" command

```
Router(config)#enable password <password>
```

   - The password is shown (not encrypted) in the configuration files (more later when introducing commands "show running-config" o "show startup-config")
   - It is possible to encrypt the password (all the passwords) using the command:

```
Router(config)#service password-encryption
```

Securing router access (4/4)

2. "**enable secret**" command

```
Router(config)#enable secret <password>
```

- The algorithm used by the *enable secret* command is more robust than the one used by the *service password-encryption* command

The "show" command (1/2)

- It is used to provide information about router features (both hardware and software)
- It is available both in the User EXEC and Privileged EXEC modes (with different information provided)
- Esempi:
  - **"show interfaces"** – statistics about interfaces
  - **"show controllers serial"** – hardware level information about interfaces
  - **"show clock"** – router clock
  - **"show hosts"** – list of devices (hostname and IP address) known by the router

The "show" command (2/2)

- **"show users"** – users connected to the router
- **"show history"** – list of commands used in the past
- "**show flash**" – information about flash memory and available IOS image files
- "**show version**" – hardware level features of the router and running IOS
- "**show ARP**" – ARP table of the router
- "**show protocol**" – Layer 3 protocols configured at router level and at interface level
- "**show startup-configuration**" – startup configuration file (saved into the NVRAM)
- "**show running-configuration**" – running configuration file (saved into the RAM)

The configuration files (1/2)

- The *running-config* file stores the actual configuration of the router
- The *startup-config* file stores the last configuration saved (that will be loaded on router re-start)
- To save the actual configuration in the NVRAM:

```
Router# copy running-config startup-config
```

FROM TO

ENTER FOR CONFIRM

The configuration files (2/2)

- If a command accepted by the router was not the one to be used it is possible to come back:
  - Execute "*no command name*"
- If all the commands executed from router startup must be removed from the configuration:
  - Reload the startup configuration file from NVRAM

```
Router# copy startup-config running-config
```

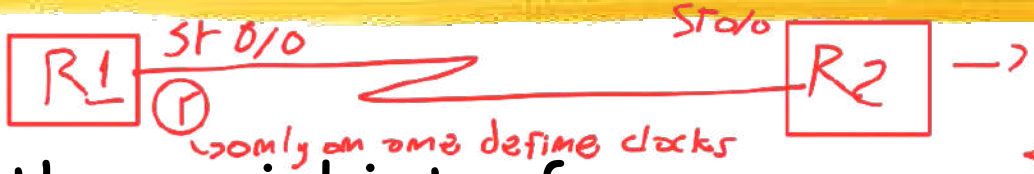
Serial interface configuration (1/2)

```
Router(config)#interface serial 0/0

Router(config-if)#ip address <ip address>
<netmask>
Router(config-if)#clock rate 56000
Router(config-if)#no shutdown
```

- The serial interfaces require a clock for synchronization purposes
- All un router interfaces are "physically down" (powered off): the powering up command is "<u>no shutdown</u>"

Ethernet interface configuration



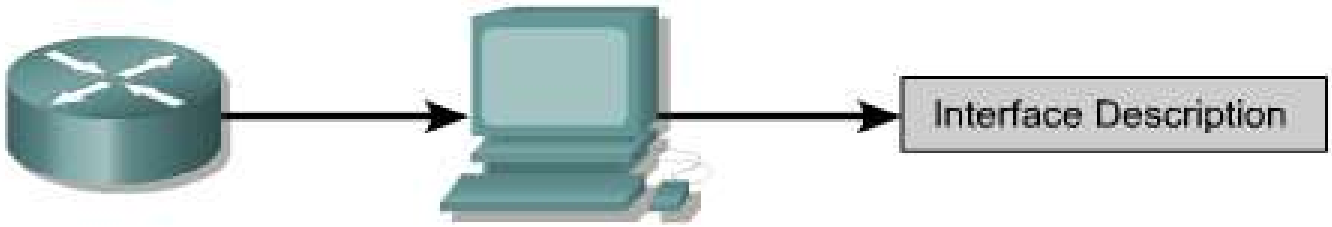
FOR ALL SERIAL INTERFACE
↓
only on one of two!

- Similar to the serial interface one.
- No need to configure a clock

Router

```
Router(config)#interface e0
Router(config-if)#ip address 183.8.126.2 255.255.255.128
Router(config-if)#no shutdown
```

Interface description (2/2)




```
Tokyo(config)#interface e 0

Tokyo(config-if)#description Engineering LAN, Bldg. 18
```

Static routing

Static routing and dynamic routing (1/2)

- The routing table of a router can be updated in two different ways:
  - Using information exchanged with different routers (Dynamic routing – Routing protocols);
  - By means of configuration commands executed by the network administrator (Static Routing)

↳ first lesson

Static routing and dynamic routing (2/2)

- The static routing is not scalable:
  - A static route must be configured manually;
  - In the case of a topology change, the network administrator must modify the proper static routes;
  - It has an high management complexity in case of big networks.
- Usually both static and dynamic routing are used

Static route configuration

· NEXT HOP ROUTER

HOW TO REACH

- The command to configure a static route is

FINAL DESTINATION

```
R1(config)#ip route Dest_IP_Add  SubNet_Mask A/B
```

0.0.0.0 0.0.0.0

ip of the interface to send packets

- Two options for the forwarding part:
  - Option A: the output interface (it is possible only in the case of a point-to-point link)
  - Option B: the next-hop router IP address
- The default route can be configured statically:

```
ip route 0.0.0.0 0.0.0.0 A/B
```

VERY FREQUENT A DEFAULT RULE

STATIC ROUTE NOT SO COMMON

The administrative distance

→ MEASURE RELIABILITY OF EACH SOURCE

- The administrative distance (AD) is a measure of the reliability degree of a route
  - Low value of AD → high reliability
- Static routes are reliable by default (AD = 1)
- It is possible to increase the administrative distance:

USE FOR CHOOSE FROM MULTIPLE ROUTE

↳ DIRECTLY CONNECTED NETWORK MOST RELIABLE SOURCE

AFTER THE ARE STATIC

AFTER THE ROUTER PROTOCO

```
ip route Dest_IP_Add  SubNet_Mask A/B AD
```

NEXT HOP IP

↳ OPTIONAL FOR CHANGING FROM DEFAULT THAT IS 1.

RELIABILITY SOURCES:
- DIR. CONN. → 0 AD
- STATIC → 1 AD
- ROUTING PROTOCOLS → >>1 AD

Routing table (1/2)

show ip rout

↳ in #

↳ enable for access

Edit Router C

Physical | Config | CLI

IOS Command Line Interface

```
Router C#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter a
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external typ
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - E
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is 172.8.1.1 to network 0.0.0.0

     172.8.0.0/24 is subnetted, 1 subnets
C       172.5.1.0 is directly connected, Serial2/0
     172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
R       172.16.0.0/16 [120/1] via 200.1.1.1, 00:00:27, Serial3/0
S       172.16.1.0/24 [1/0] via 200.1.1.1
     172.64.0.0/24 is subnetted, 1 subnets
C       172.64.1.0 is directly connected, FastEthernet0/0
C    200.1.1.0/24 is directly connected, Serial3/0
S*   0.0.0.0/0 [1/0] via 172.8.1.1
Router C#
```

NOT REPORTED BUT IS O.

m → A.D.

Copy | Paste

Directly Connected Route → C

Dynamically Updated Route → R

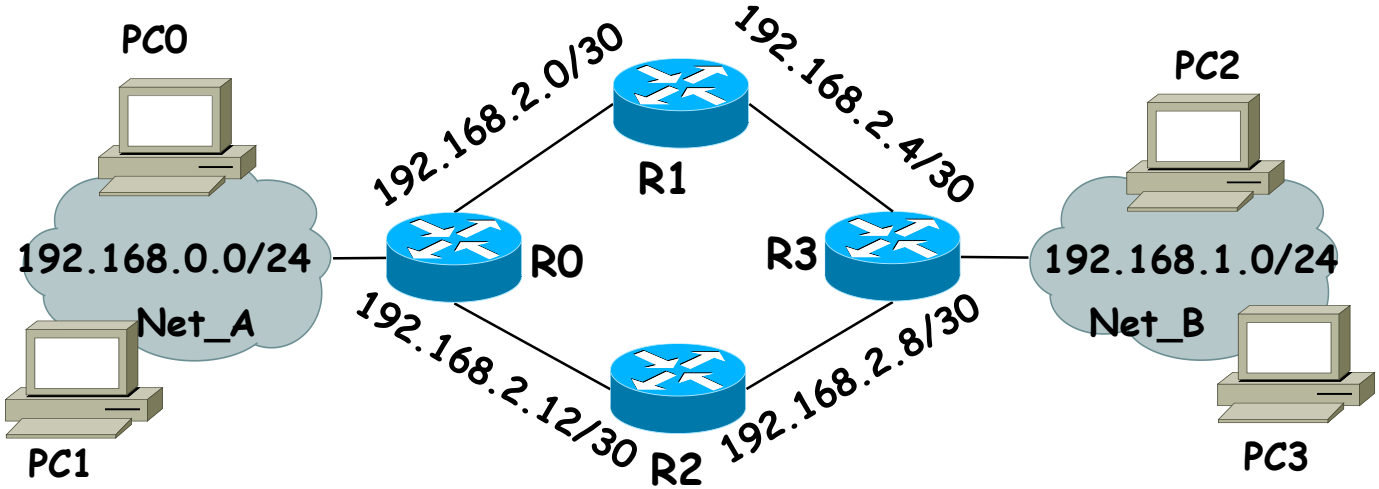
Static Route → S

Default Route → S*

Routing table (2/2)

- The rows (route) of a routing table have different sources:
  - Directly connected routes (C)
  - Statics routes (S,S*) default route
  - Dynamic Routes (RIP→R, OSPF→O, etc…)
- The administrative distance is a measure of the route reliability (depends on the source)
- To check reachability: **ping** and **traceroute**

Practical activity



- DIRECTLY CONNECTED NETWORK ARE THE MORE RELIABLE ONE SOURCES

192.168.0.0/24

11000000 . 10101000 . 00000000 . xx xxxxxx

192.168.0.0 ← Network add.

192.168.0.1

.....

192.168.0.254

192.168.0.255 ← Broadcast add.

only usable

192.168.2.4/30

11000000.10101000.00000010.000001xx

192.168.2.4 ← Network add.

192.168.2.5

192.168.2.6

192.168.2.7 ← Broadcast add.

Antonio Cianfrani

Routing Protocols

Routing protocols

- A routing protocol provides a communication channel among routers to exchange reachability information about networks
- Routing tables are properly configured and immediately updated
- List of routing protocols:
  - ✓ Routing Information Protocol (RIP)
  - ✓ Enhanced Interior Gateway Routing Protocol (EIGRP)
  - ✓ Open Shortest Path First (OSPF)
  - ✓ Border Gateway Protocol (BGP)

→ change in base of communication, but each one is based on exchange of reachability information between routers

→ FOR COMMUNICATION BETWEEN AS

RIP – OSPF ⇒ INTERIOR – GATEWAY PROTOCOLS

Convergence

→ SPEED OF REACTING OF CHANGING IN REACHABILITY

- The performance of a routing protocol is the convergence speed
- A routing protocol reaches the convergence state when all the network routers have the same network view
- A fast convergence is required since during convergence a router can have a wrong routing table (and so forwarding decisions could lead to packet loss)

RIP → SLOW TO REACH CONVERGENCE

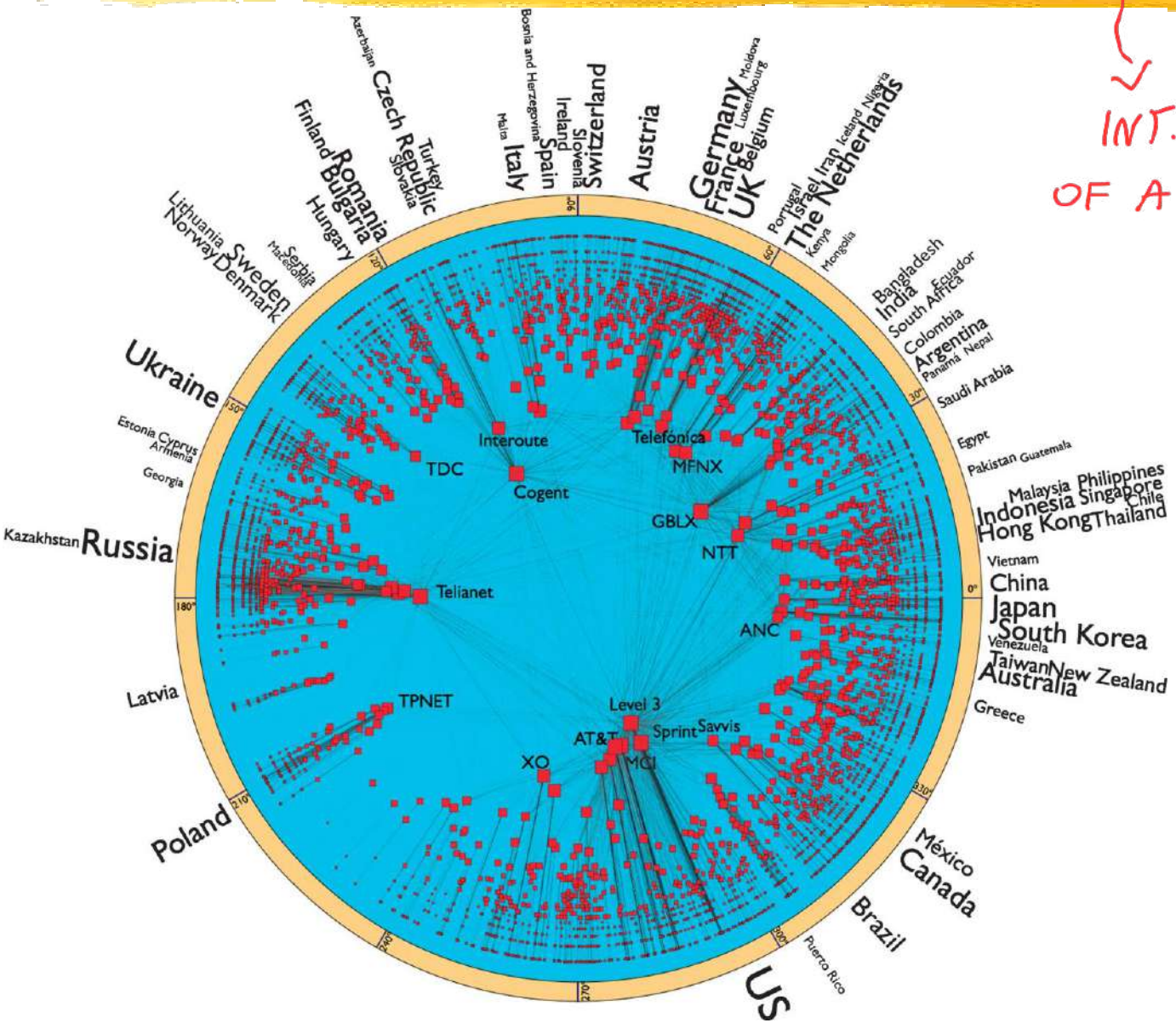
OSPF → FAST BUT MORE COMPLEX

Autonomous System ⇝ like TIM, VODAFONE, ...

- An Autonomous System (AS) is a set of networks and routers managed by the same network administrator
- An AS is seen from outside as a single entity
  - An AS is identified by a single number (16 bit - AS number)
- Internet is composed of a set of ASes. Each AS manages its own set of networks:
  - There will be an independent routing protocol for the routing inside each AS
  - Routing among ASes must be the result of an agreement among ASes.

Internet map with ASes

INT. IS A SET OF AS.



Routing protocols classification

- Routing protocol classification on the basis of functioning principles:
  - Distance vector: RIP, BGP
  - Link state: OSPF

BOTH ARE WAYS TO COMPUTE NETWORK PATHS, BUT DO IN DIFFERENT WAY.

- On the basis of use scenario:
  - Interior Gateway Protocol (IGP): intra-AS (RIP and OSPF)
  - Exterior Gateway Protocol (EGP): inter-AS (BGP)

Antonio Cianfrani

Distance Vector Routing protocols

Distance vector routing protocols (1/2)

↳ HIGH. SCALABLE : ONLY WITH NEIGHBORS

Main features of a distance vector routing protocol :

- Each router exchanges routing information only with neighbor routers (DIRECTLY CONNECTED ROUTERS)
  - Two routers are neighbors if they are directly connected by means of a network
- Each router sends to the neighbors routers messages reporting its own routing table
- Routing messages are sent periodically (the time period is a protocol parameter: in the case of RIP the default value is 30 seconds)
- Each row of the routing table sent has its own distance (depends on the protocol metric: in the case of RIP the metric is the number of hops)

→ number of hops in RIP

Distance vector routing protocols (2/2)

- In the distance vector case, a router doesn't have a global network view → KNOW ONLY VIEW OF NEIGHBORS
- A router computes its own routing table:
  - for each route two information are stored: distance (depends on the metric) and next-hop router
  - the directly connected routes have a distance equal to 0
- When a message is received (routing table of a neighbor), an algorithm is applied (Bellmann-Ford one for RIP) to update the routing table if new better routes (on the basis of the distance) are available.

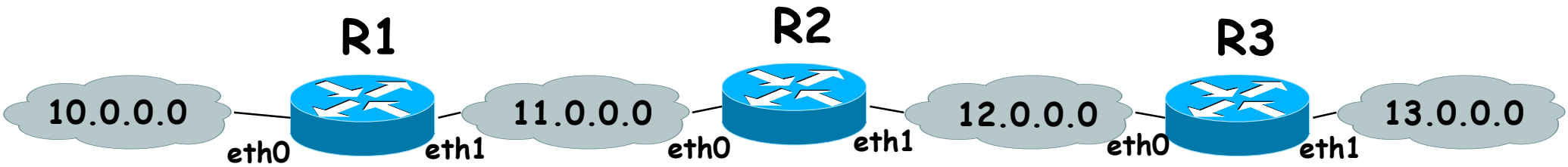
Example (1/6)

A.D.

number of hop

[120/1]

*/8 the netmask is the same for each LAN



R1

Network	Next-hop	Distance

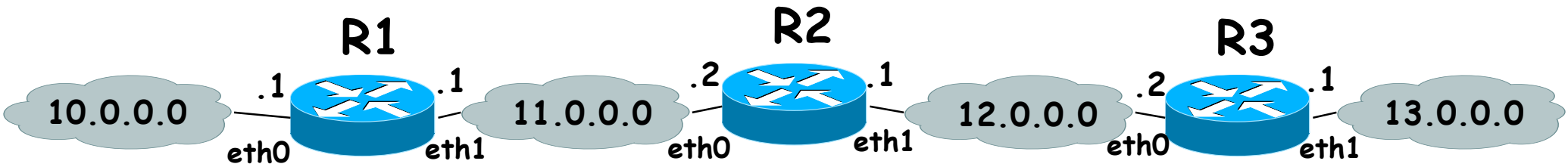
R2

Network	Next-hop	Distance

R3

Network	Next-hop	Distance

Example (2/6)



R1

Network	Next-hop	Distance
10.0.0.0	eth0	0
11.0.0.0	eth1	0
	...	

WE WRITE
directly connected
by means of eth#

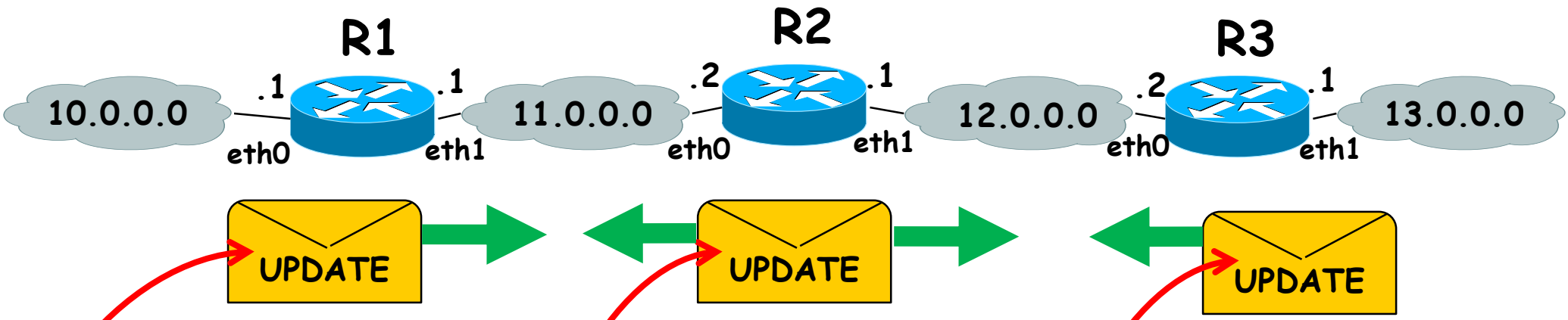
R2

Network	Next-hop	Distance
11.0.0.0	eth0	0
12.0.0.0	eth1	0

R3

Network	Next-hop	Distance
12.0.0.0	eth0	0
13.0.0.0	eth1	0

Example (3/6)



R1

Network	Next-hop	Distance
10.0.0.0	eth0	0
11.0.0.0	eth1	0

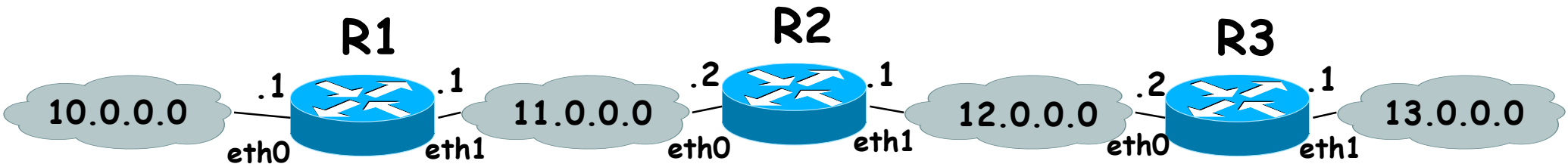
R2

Network	Next-hop	Distance
11.0.0.0	eth0	0
12.0.0.0	eth1	0

R3

Network	Next-hop	Distance
12.0.0.0	eth0	0
13.0.0.0	eth1	0

Example (4/6)



R1

Network	Next-hop	Distance
10.0.0.0	eth0	0
11.0.0.0	eth1	0
12.0.0.0	11.0.0.2	1

R2(0) + 1

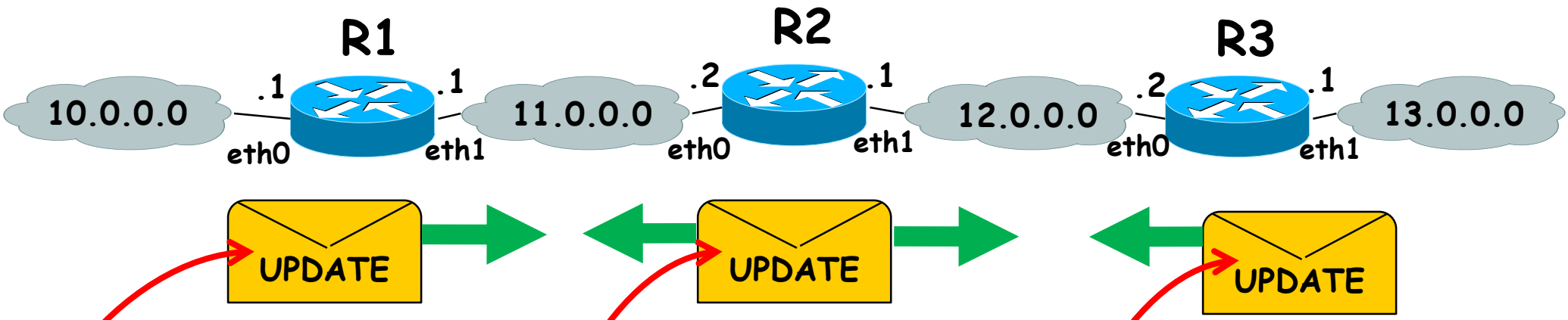
R2

Network	Next-hop	Distance
11.0.0.0	eth0	0
12.0.0.0	eth1	0
10.0.0.0	11.0.0.1	1
13.0.0.0	12.0.0.2	1

R3

Network	Next-hop	Distance
12.0.0.0	eth0	0
13.0.0.0	eth1	0
11.0.0.0	12.0.0.1	1

Example (5/6)



R1

Network	Next-hop	Distance
10.0.0.0	eth0	0
11.0.0.0	eth1	0
12.0.0.0	11.0.0.2	1

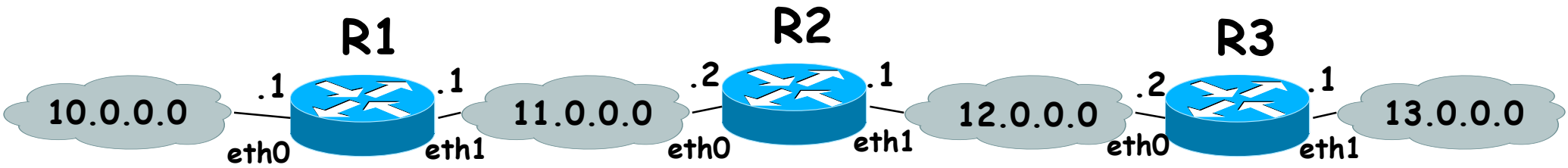
R2

Network	Next-hop	Distance
11.0.0.0	eth0	0
12.0.0.0	eth1	0
10.0.0.0	11.0.0.1	1
13.0.0.0	12.0.0.2	1

R3

Network	Next-hop	Distance
12.0.0.0	eth0	0
13.0.0.0	eth1	0
11.0.0.0	12.0.0.1	1

Example (6/6)



R1

Network	Next-hop	Distance
10.0.0.0	eth0	0
11.0.0.0	eth1	0
12.0.0.0	11.0.0.2	1
13.0.0.0	11.0.0.2	2

R2

Network	Next-hop	Distance
11.0.0.0	eth0	0
12.0.0.0	eth1	0
10.0.0.0	11.0.0.1	1
13.0.0.0	12.0.0.2	1

R3

Network	Next-hop	Distance
12.0.0.0	eth0	0
13.0.0.0	eth1	0
11.0.0.0	12.0.0.1	1
10.0.0.0	12.0.0.1	2

REACH CONVERGENCE
AFTER THREE EXCHANGE

Antonio Cianfrani

Routing Information Protocol (RIP)

RIP protocol

ROUTING INFORMATION PROTOCOL

only used in a small network scenario

- RIP is a distance vector routing protocol
- The number of hops (number of routers crossed) is used as distance to detect the best path

  distance

  max lenght acceptable is 15
- A distance equal to 16 means destination unreachable
- Routing updates are sent in broadcast every 30 seconds
- RIP messages are encapsulated into UDP packets, with source and destination ports both equal to 520
- The administrative distance of RIP is 120

RIPv1

- RIP versions
  - Classful Routing Protocol (RIP v1)
  - Classless Routing Protocol (RIP v2) ⇝ more easy to configure
- In the RIPv1 messages the subnet mask is not provided.
- How to choose the subnet mask?: two methods:
  - Using the default one on the basis of the classful approach (classes A,B or C)
  - The subnet mask of the network connected to the RIP message incoming interface is used
- RIPv1 is defined classful but it works even if subnetting is used IF
  - subnetting is performed with fixed subnet length AND
  - the subnetted networks are contiguous. ⇝ NOT VARIABLE SUBNETTING IN MAJOR NETWORK AND CONTINUOS!

RIPv1 configuration (1/2)

global configuration mode

- The "router rip" command enables RIP
- The command "network" is used to define the interfaces running RIP:
  - the interfaces running RIP will send and receive RIP messages
  - the networks (directly) connected to interfaces running RIP will be inserted (as routes) in the RIP messages sent

major network is 10.0.0.0/8

10.0.1.0 → in classful class A → /8

must compare major network of classfull approach with one of LAN

192. is class C → /24

STRONGLY DEPEND BY IP RECIVED AND THAT OF LAN

192.168.0.0

NETMASK? → /25

can not be /24 because major 192.168.0.0/24 would conflict with LAN

R1, R2 DIRECTLY CONNECTED

192.168.0.0, d=2 ⇝ netmask assigned by R2?

R1 —RIP→ R2

↳ in v1 no netmask field

but can also work in no classful approach but with some constraint

how to detect netmask?

Configurazione del protocollo RIPv1 (2/2)

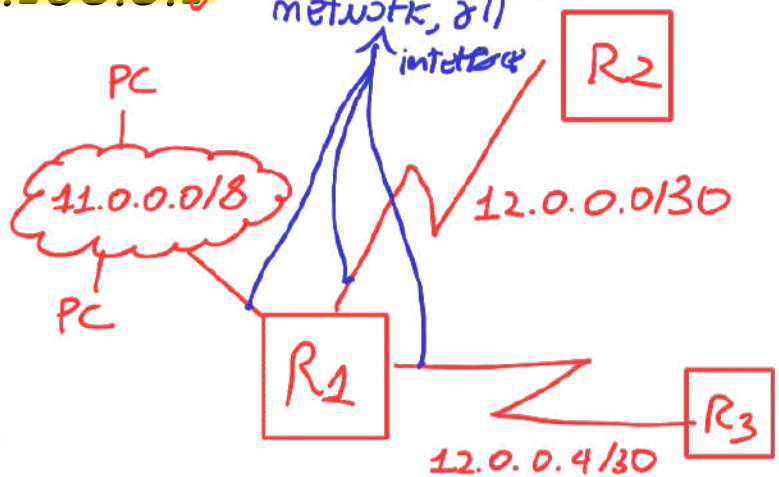
- If the network address is "wrong" the router is able to perform the correction

if we doesn't configure some of it, them won't be in RIP messages

```
R3(config)#router rip
R3(config-router)#network 192.168.4.0
R3(config-router)#network 192.168.5.0
```

must configure RIP in all directly connected network, all interface

```
R3#show running-config
!
router rip
network 192.168.4.0
network 192.168.5.0
!
```



Want configure R1 with RIP

in this case 12.0.0.0 would be added → not 12.0.0.0 12.0.0.4

RIP and the routing table

- RIP route in the routing table (*show ip route*)

R 192.168.205.0/24 [120/3] via 151.100.37.8, 00:00:04, Eth0

- RIP protocol
- address and netmask of the destination network
- administrative distance
- number of hops
- next-hop router
- times from the last RIP message (the next one in 26 s)
- output interface toward the destination network

RIP timers

- **Invalid timer**. If a route is not updated after "invalid timer" seconds (default value 180 s), it is flagged as invalid and its distance is set to 16.
- **Flush timer**. If a router is not updated after "flush timer" seconds (default value 240 s), it is removed from the routing table.

show ip protocols

```
R2#show ip protocols
Routing Protocol is "rip"
  Sending updates every 30 seconds, next due in 23 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Redistributing: rip
  Default version control: send version 1, receive any version
    Interface             Send  Recv  Triggered RIP  Key-chain
    FastEthernet0/0       1        1 2
    Serial0/0/0           1        1 2
    Serial0/0/1           1        1 2
  Automatic network summarization is in effect
  Maximum path: 4
  Routing for Networks:
    192.168.2.0
    192.168.3.0
    192.168.4.0
```

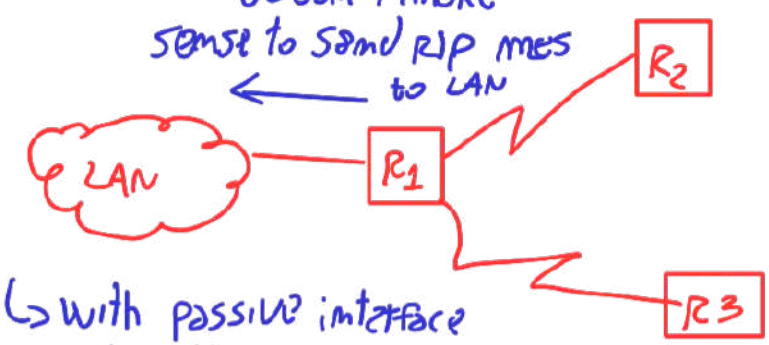
Passive-interface

- The command *passive-interface* is used to avoid sending RIP messages through a specific interface (and so into a network)
- The router still inserts the network connected to the passive interface in the RIP messages.

```
# router rip
(....) # passive-interface Fa 0/0
Default version control: send version 1, receive any version
    Interface             Send  Recv  Triggered RIP  Key-chain
    Serial0/0/0           1     1 2
    Serial0/0/1           1     1 2
Automatic network summarization is in effect
Routing for Networks:
    192.168.2.0
    192.168.3.0
    192.168.4.0
Passive Interface(s):
    FastEthernet0/0
```

but LAN will be inserted to mes. to R_2 and R_3

doesn't make sense to send RIP mes to LAN



↳ with passive interface we doesn't send to it

RIP debug

```
R2#debug ip rip
RIP protocol debugging is on
RIP: received v1 update from 192.168.2.1 on Serial0/0/0
      192.168.1.0 in 1 hops
RIP: received v1 update from 192.168.4.1 on Serial0/0/1
      192.168.5.0 in 1 hops
RIP: sending  v1 update to 255.255.255.255 via FastEthernet0/0 (192.168.3.1)
RIP: build update entries
      network 192.168.1.0 metric 2
      network 192.168.2.0 metric 1
      network 192.168.4.0 metric 1
      network 192.168.5.0 metric 2
RIP: sending  v1 update to 255.255.255.255 via Serial0/0/1 (192.168.4.2)
RIP: build update entries
      network 192.168.1.0 metric 2
      network 192.168.2.0 metric 1
      network 192.168.3.0 metric 1
```

Antonio Cianfrani

RIPv2

RIPv1

- RIP v1: Classful Routing Protocol

- In RIPv1 the subnet mask of destination networks is not specified in the RIP messages:
  1) **How to detect the subnet mask?**
  2) **How to face subnetting?**

- Major Network concept: class A, B o C address associated to an IP address.

  Example: major network of 192.168.0.128/25 is 192.168.0.0/24

remember!

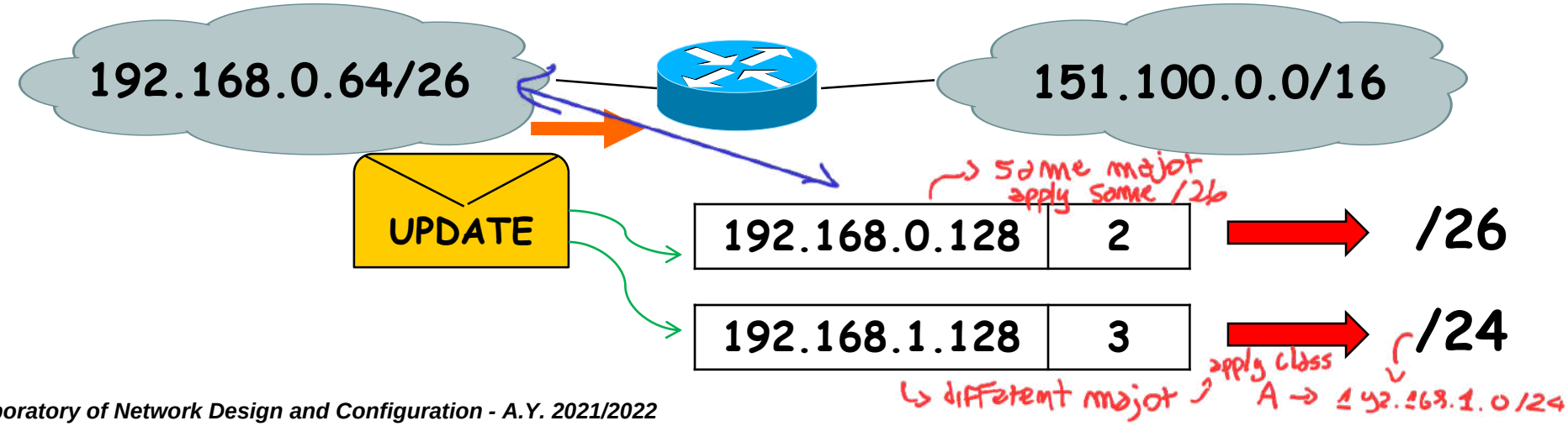
Address Class	1st octet range (decimal)
A	1-127** /8
B	128-191 /16
C	192-223 /24

RIP messages processing

- If a router receives an Update (RIP message) containing a destination network with IP address A.B.C.D, it associates the subnet mask following two rules:

  1) If the destination network has a major network different than the one of the network directly connected to the received interface, then the classful subnet mask is used.

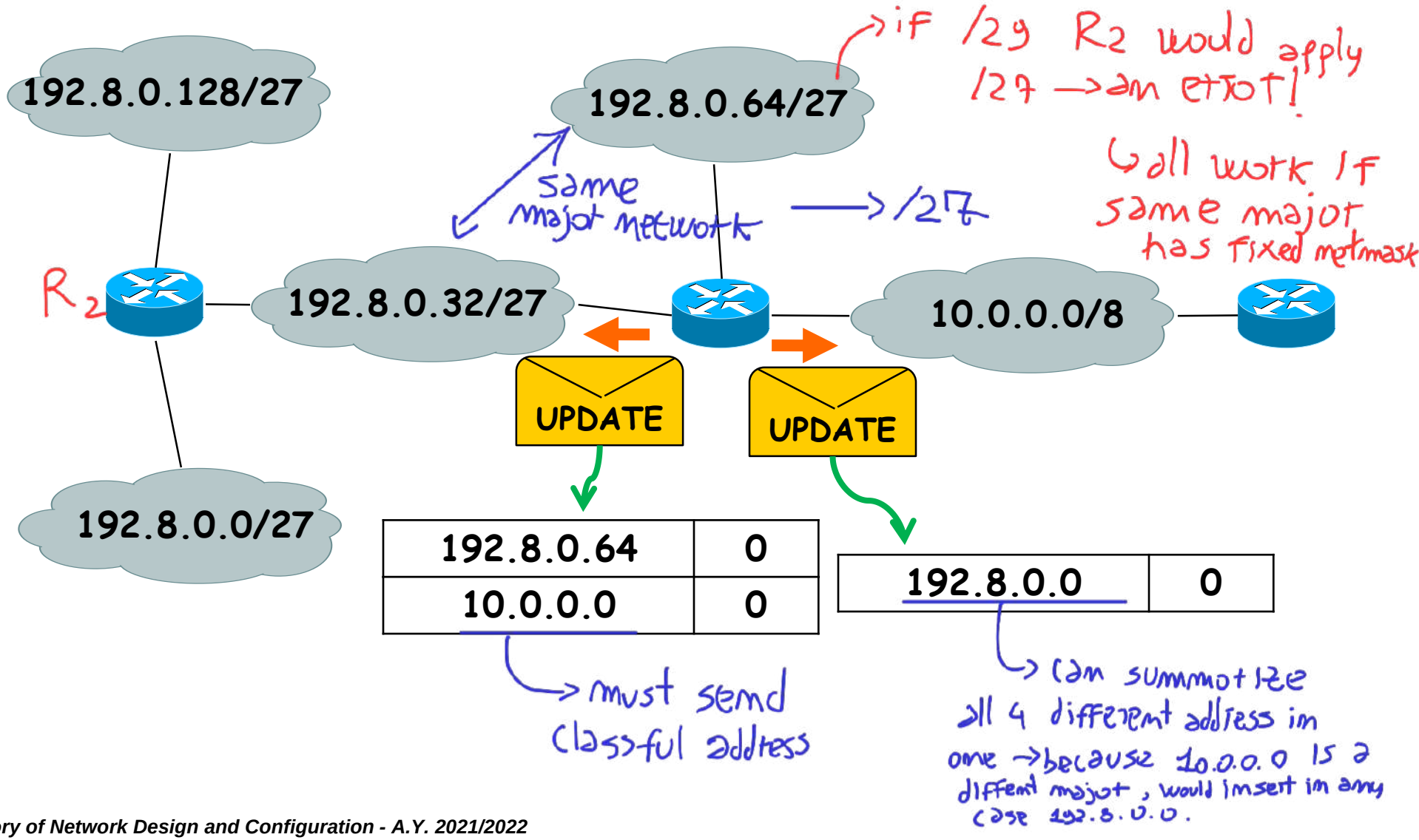
  2) If the destination network has a major network equal to the one of the network directly connected to the received interface, then the same subnet mask of the interface is used.



RIP messages generation

- If a router has to send a RIP message through an outgoing interface and the destination IP address of the route is obtained by subnetting:
  1) If the destination address has a major network different than the one of the network directly connected to the interface, then the IP address of the major network is used: **summarization**
  2) If the destination network has a major network equal to the one of the network directly connected to the interface, then the same subnet mask of the interface is used.

Summarization: example



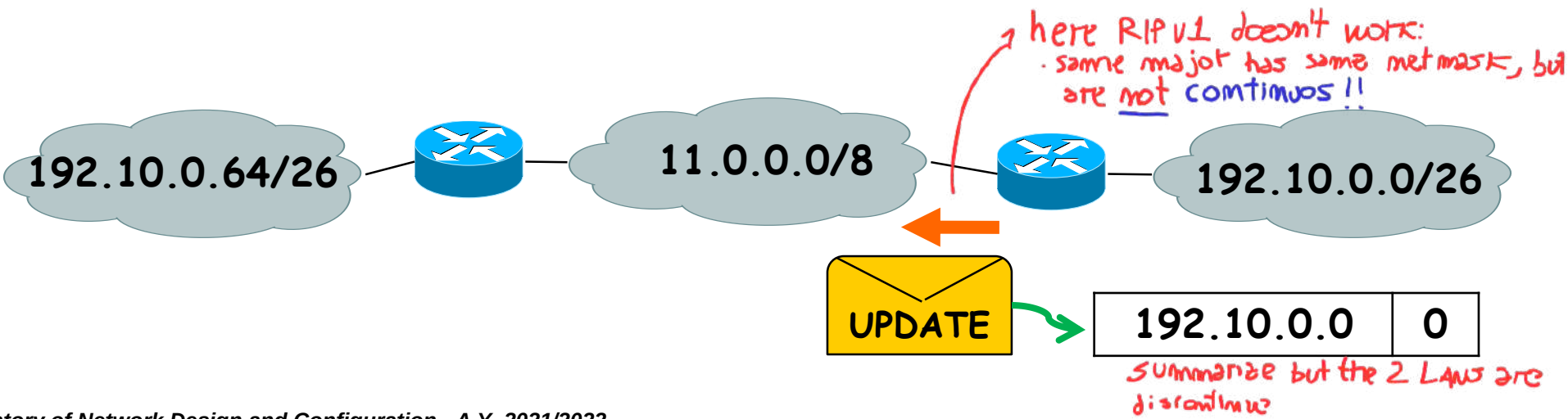
Summarization

Advantages:

- Low number of RIP Messages
- Shorter routing table

Weak aspects:

- Subnetting with variable mask length is not possible
- Networks with same major network must be contiguous



RIPv2

- RIP v2: Classless Routing Protocol
- In the RIPv2 messages the subnet mask field is present:
- The configuration commands:

R3(config)# router rip

R3(config-router)# version 2

- In the case of variable or discontinuous subnetting, RIPv2 doesn't work!

WHY?

RIPv2: auto-summary

- Let's take a look to show ip protocols

```
R1#show ip protocols
Routing Protocol is "rip"
  Sending updates every 30 seconds, next due in 20 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Redistributing: rip
  Default version control: send version 2, receive version 2
    Interface             Send  Recv  Triggered RIP  Key-chain
    FastEthernet0/0       2     2
    FastEthernet0/1       2     2
    Serial0/1/0           2     2
  Automatic network summarization is in effect
  Maximum path: 4
  Routing for Networks:
    172.30.0.0
```

- The option *Automatic network summarization* is automatically enabled: RIPv2 acts as RIPv1.

- In the case of variable or discontinuous subnetting, it must be deactivated:

R3(config-router)# no auto-summary

Default router distribution → we configure only border router

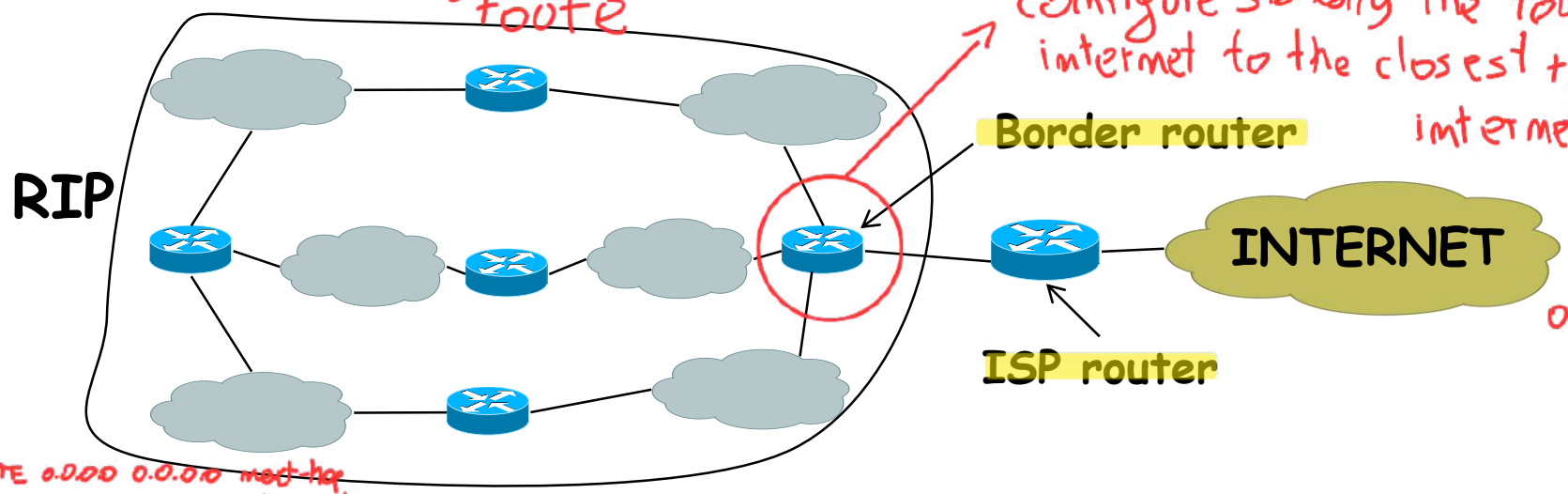
- It is possible to configure the default router only on the border router and then "to distribute" it to remaining routers using RIP:

for share Internet path to all other with RIP

Router (config-router)# default-information originate

only default route

configure statically the route to internet to the closest router to internet



only in one router!

1st IP ROUTE 0.0.0.0 0.0.0.0 next-hop
2nd def ... -inf.~ originate

- Command to distribute static routes:

all other static redistribute

Router (config-router)# redistribute static

Antonio Cianfrani

Open Shortest Path First (OSPF)

The OSPF protocol

most used one

→ more reliable

A.D. = 110 < 120

- OSPF is an intra-AS routing protocol (IGP).
- OSPF is a link-state routing protocol: each network router knows the network topology.
- The OSPF packets are directly encapsulated into IP packets, with protocol field equal to 89.
- The OSPF administrative distance is equal to 110.

→ A.D. = 110 , RIP was 120

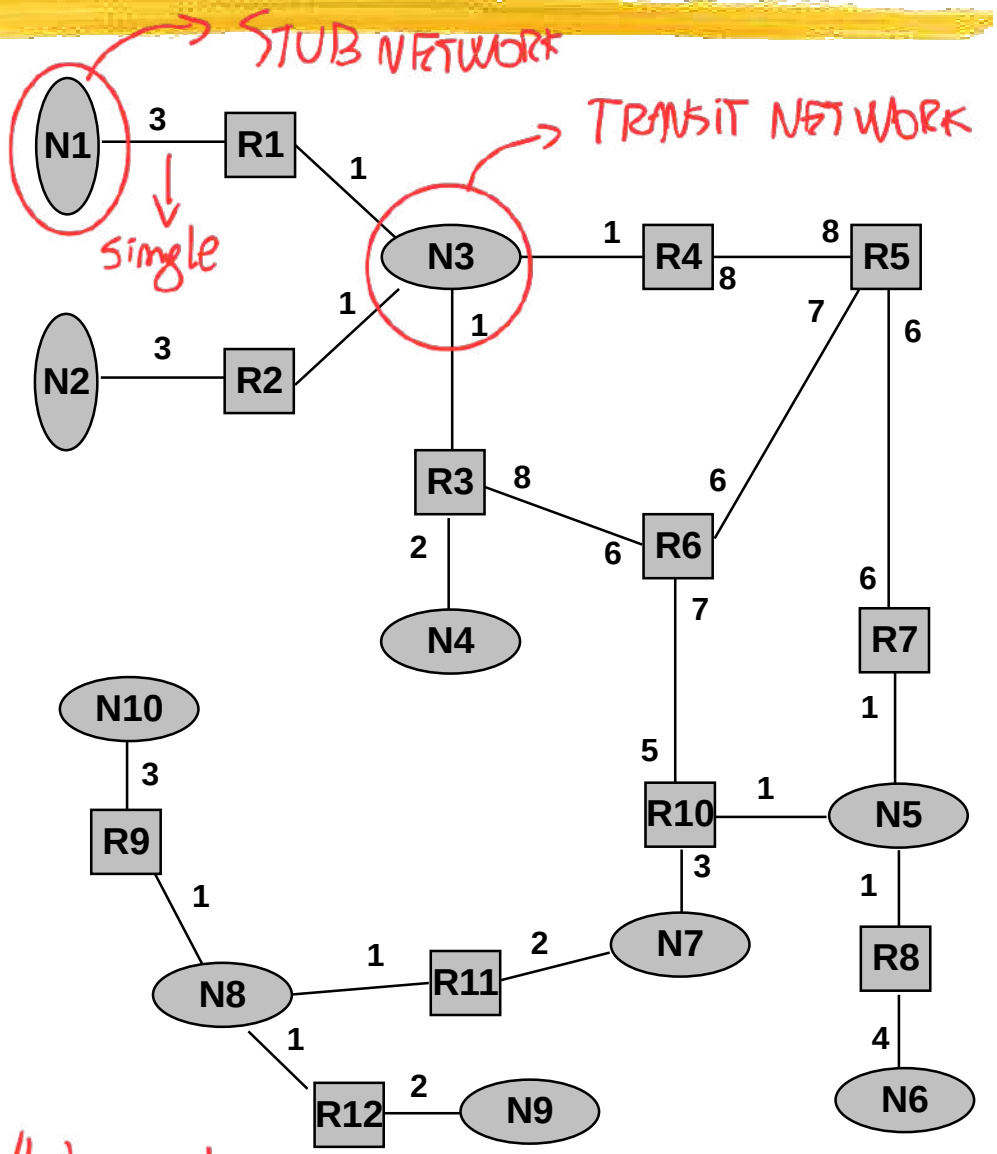
Functioning principles

- Each router periodically send (default: 10 s) *Hello packets*, to notify its links status to the neighbors
- Each router describes its topology generating a Link State Advertisement (LSA) packet. → insert number of interfaces and neighbours.
- The LSA is sent to all network routers using the flooding technique.
- Each router has a global network view (LSA database) and it is able to compute the set of shortest paths (Shortest Path Tree: SPT) executing the Dijkstra algorithm.

each router receive all LSA packet, know exactly network Graph → can compute all shortest path

Routers and networks

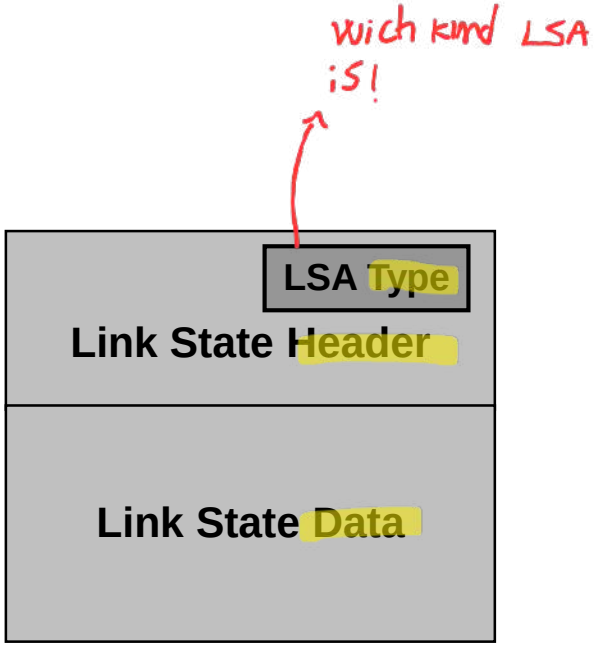
- The elements of an AS are:
  - Routers
  - Networks: classified in Transit Networks (more than one router) and Stub Networks (connected to a single router)



- Stub network not very interested
- Need that TRANSIT NETWORK is well defined

Link State Advertisment (LSA)

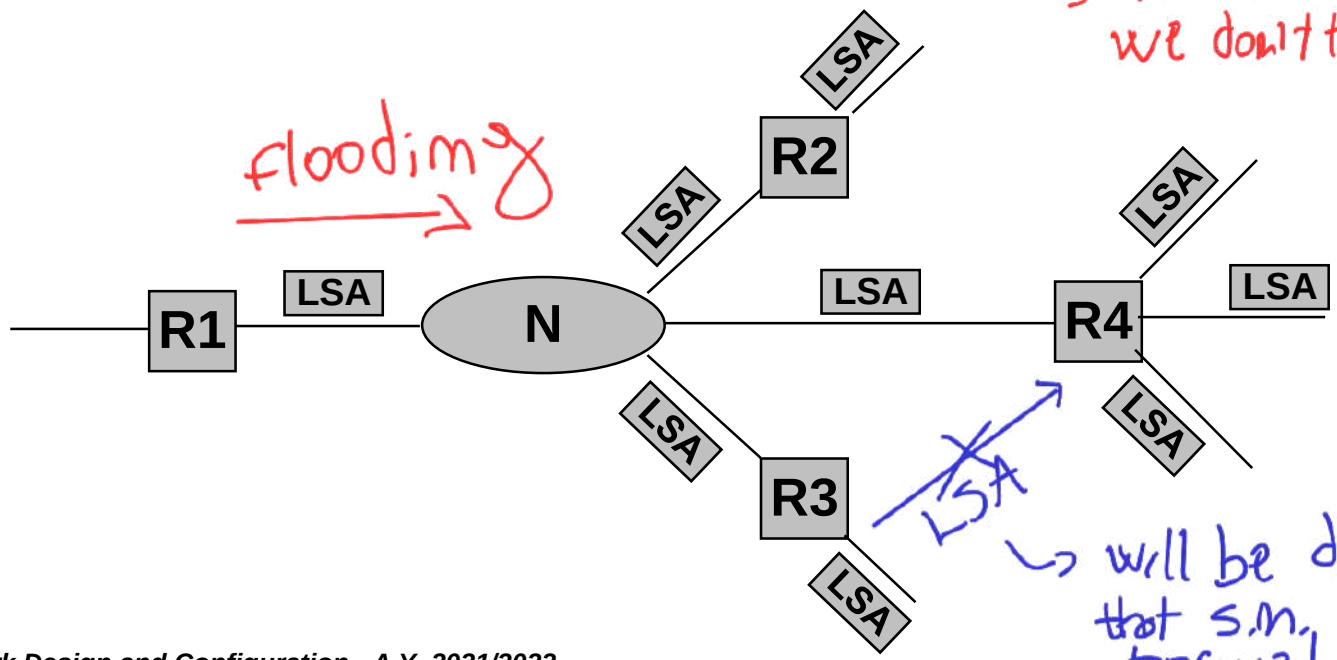
- An LSA is composed of a Link State Header and a Link State Data
- The Type field identifies the different LSAs:
  - ✓ Router_Lsa (type 1)
  - ✓ Network_Lsa (type 2)
  - ✓ ……



Flooding

- The *flooding* technique:
  - an LSA received on an incoming interface is forward to all the remaining interfaces
  - each LSA has a sequence number
    - ✓ To avoid the forwarding of "old" LSAs (already forwarded);
    - ✓ To update the LSA database only if needed.

↳ if we have a cycle we don't resend the LSA.



↳ will be discarded because that s.n. of LSA is already recived

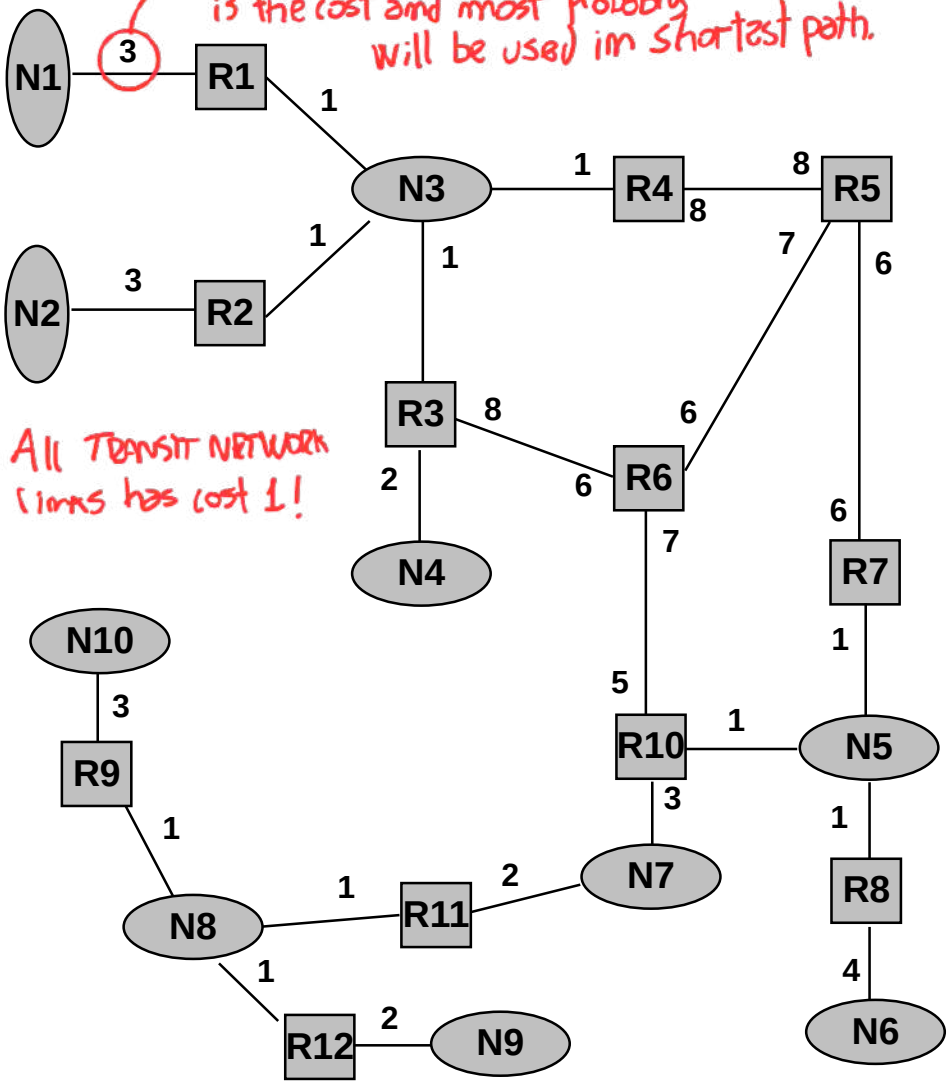
Shortest Path Tree computation

- A router maintains a database with the latest data on network topology.
- The network topology is a weighted graph where:
  - Nodes are routers and networks
  - Links are
    - Point-to-point links among routers;
    - Links among a router and a network.
- Each router computes the Shortest Path Tree (Dijkstra algorithm) and it is able to update its routing table.

OSPF: Example (1/2)

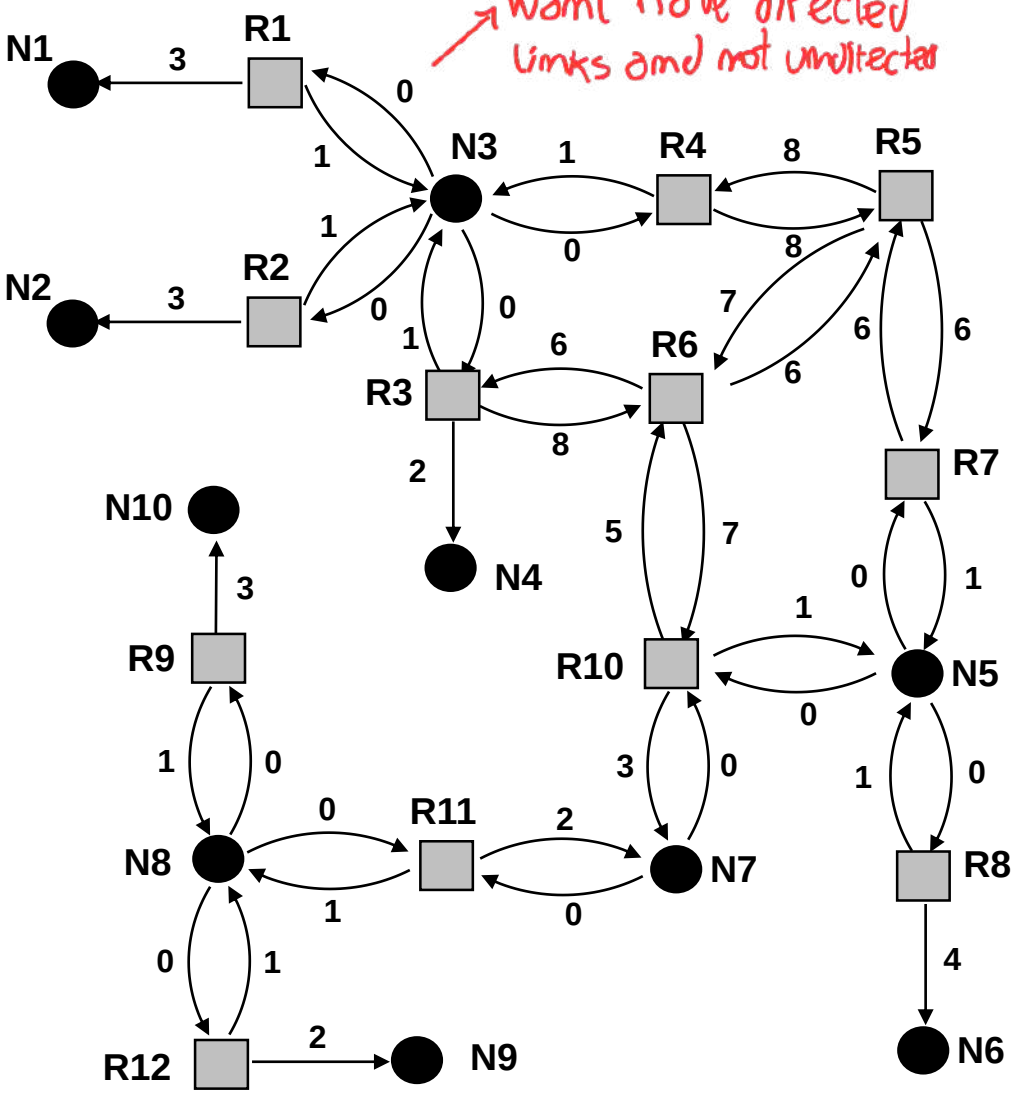
· TRANSIENT NETWORK 1 R→N, 0 : N→R , STUB NETWORK R→N

cost for each link, lower is the cost and most probably will be used in shortest path.



· All TRANSIT NETWORK links has cost 1!

want have directed links and not undirected

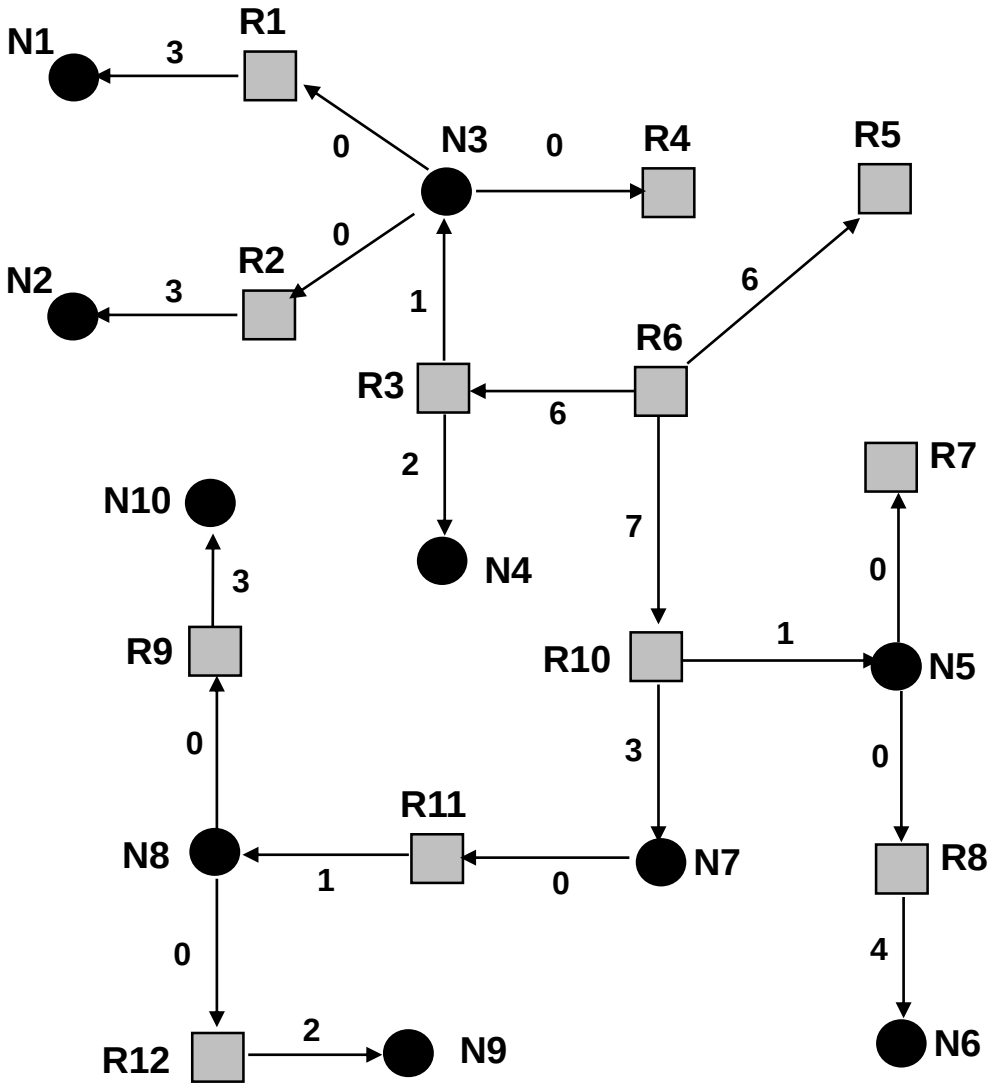


OSPF: Example (2/2)

- Shortest Path Tree e Routing Table for router R6

Destin.	N.H.	Cost
N1	R3	10
N2	R3	10
N3	R3	7
N4	R3	8
N5	R10	8
N6	R10	12
N7	R10	10
N8	R10	11
N9	R10	13
N10	R10	14

only networks



OSPF configuration

can enable passive-interface like RIP

- The command to access to OSPF configuration is:

R(config)# router ospf *id*

way for identify OSPF process

- The *id* value is a number in interval [1 – 65535] and it has a local validity.

- Interfaces (and so networks) on which enable OSPF:

R(config-router)# network *network-address wildcard-mask*
area *area-number*

  - The wildcard mask is obtained from netmask inverting 0s and 1s
  - Area: an AS can be divided into areas to solve scalability issues; area 0 is always present.

Wildcard-mask: opposite of netmask

255.255.255.0
0.0.0.255

255.255.255.192
0.0.0.63

63
001....1
110....0

OSPF link cost

- The OSPF link cost it he metric value used to perform path computation.
- Can be changed into the interface configuration section:

R(config)# interface FastEthernet X/Y

R(config-if)# ip ospf cost *cost*

- By default it depends on the bandwidth:

Interface Type	10^8/ bps = Cost
Fast Ethernet and faster	10^8/ 100,000,000 bps = 1
Ethernet	10^8/ 10,000,000 bps = 10
E1	10^8/ 2,048,000 bps = 48
T1	10^8/ 1,544,000 bps = 64
128 kbps	10^8/ 128,000 bps = 781
64 kbps	10^8/ 64,000 bps = 1562
56 kbps	10^8/ 56,000 bps = 1785

- It can be used to implement simple Traffic Engineering

Verifica configurazione OSPF

R# show ip ospf neighbor → directly connected routers

```
R1#show ip ospf neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
10.3.3.3          1   FULL/  -        00:00:30    192.168.10.6    Serial0/0/1
10.2.2.2          1   FULL/  -        00:00:33    192.168.10.2    Serial0/0/0
```

↳ FULL: everything work properly

R# show ip protocols

```
R1#show ip protocols
Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 10.1.1.1
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    172.16.1.16 0.0.0.15 area 0
    192.168.10.0 0.0.0.3 area 0
    192.168.10.4 0.0.0.3 area 0
 Reference bandwidth unit is 100 mbps
  Routing Information Sources:
    Gateway         Distance      Last Update
    10.2.2.2             110      11:29:29
    10.3.3.3             110      11:29:29
  Distance: (default is 110)
```

we can redistribute default and static route like in RIP

R# show ip ospf database

Antonio Cianfrani

Multi-area OSPF

OSPF scalability issues

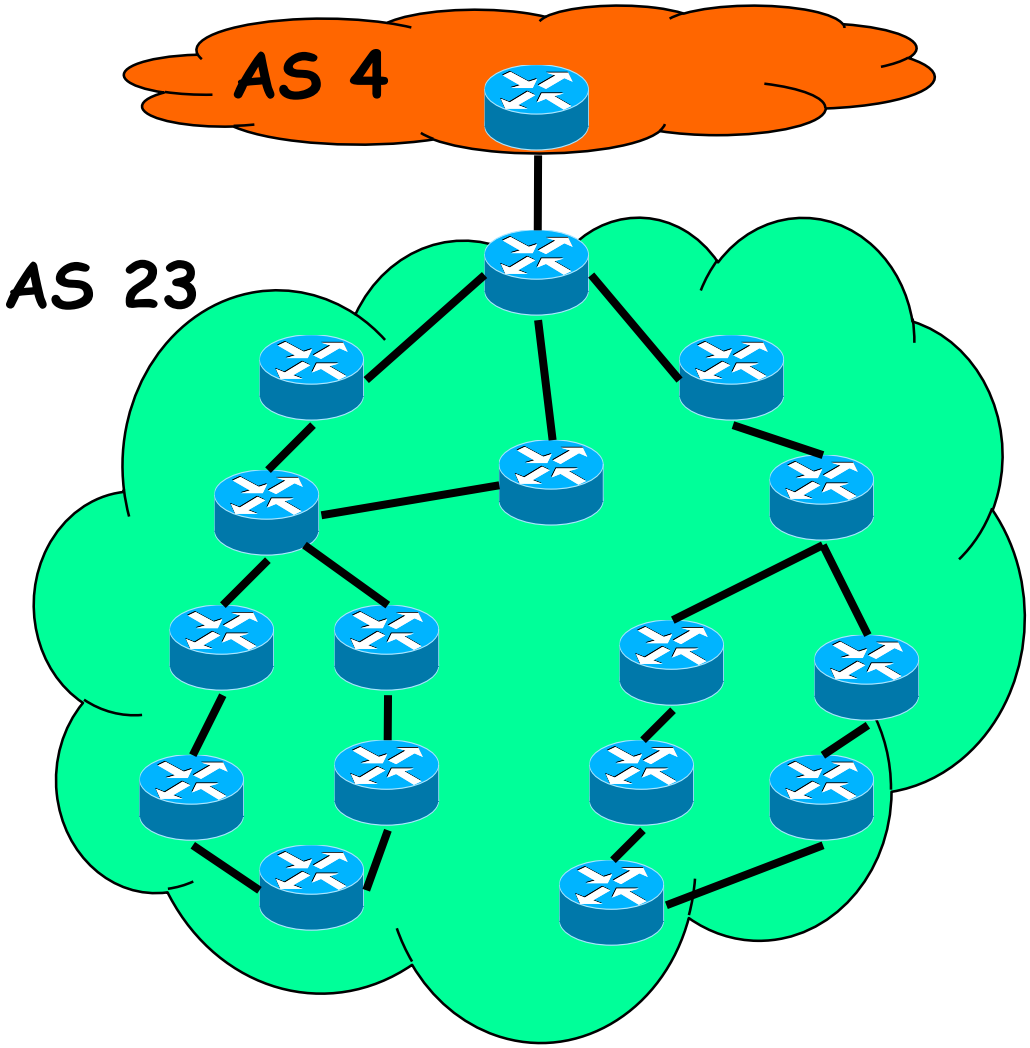
→ database can be really huge

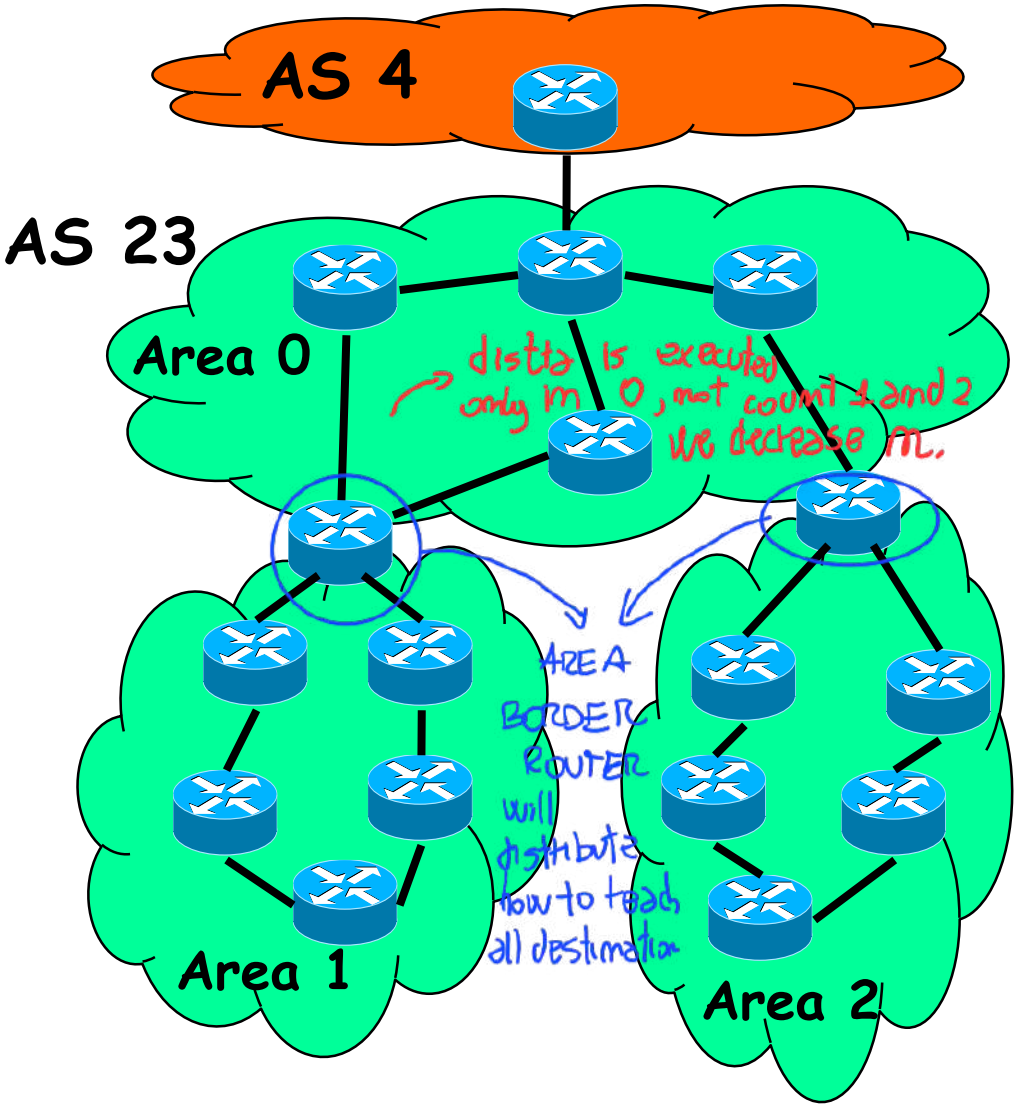
- The OSPF protocol can have scalability issues:
  - Each router maintains a database with the whole set of LSAs
  - The LSAs of routers and transit networks are sent inside the AS
  - The Shortest Path Tree computation (Dijkstra algorithm) time depends on the number of routers and transit networks → O(NlogN)

↳ compute Dijkstra every topological change

- The AS is divided into areas
- In each area, the "internal" routers exchange LSAs:
  - Different Databases in different areas
  - Flooding is restricted to the area

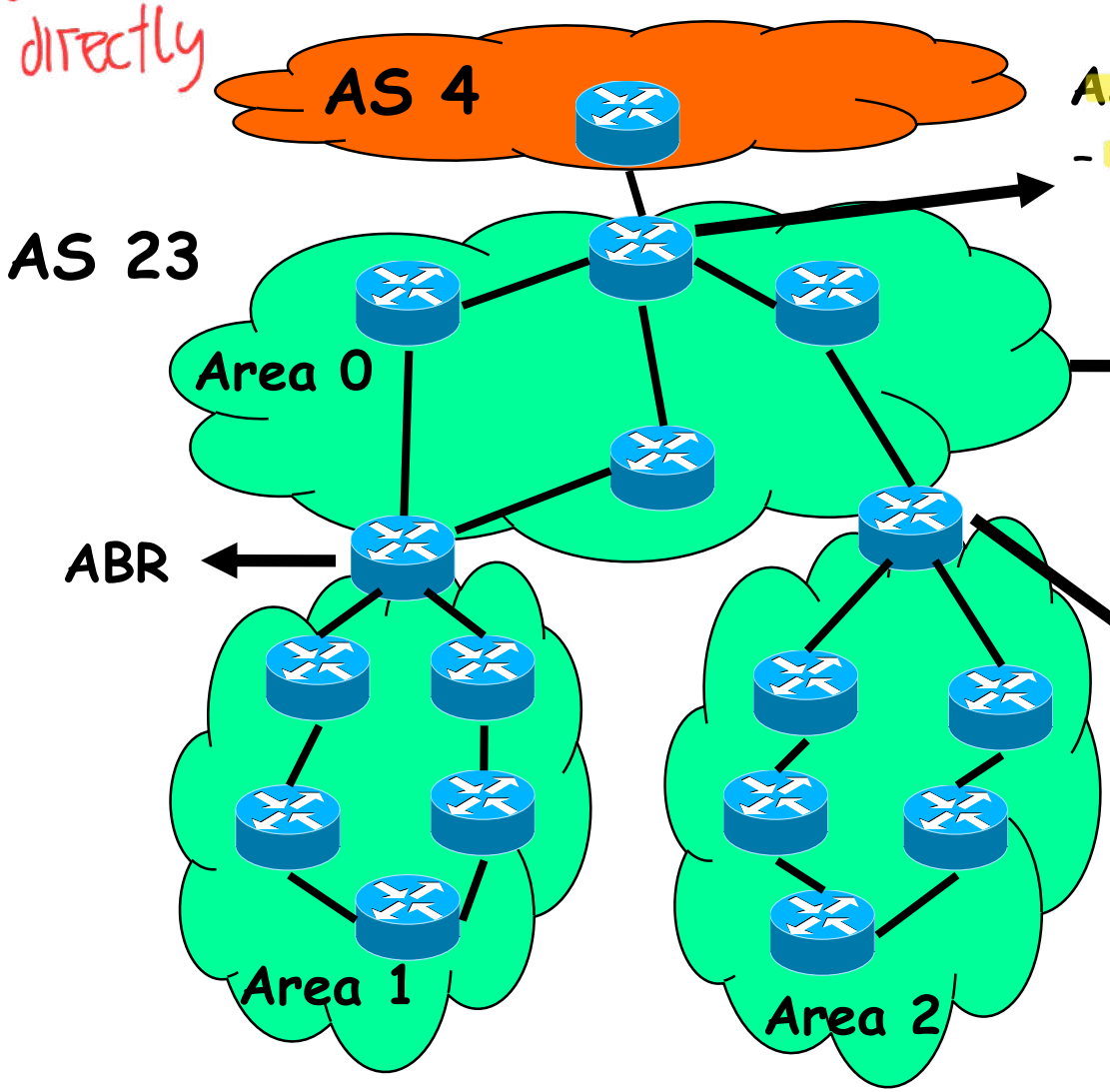
Multi-area OSPF (1/2)





ALWAYS EXIST area 0, must be connected to all areas directly

Multi-area OSPF (2/2)



ASBR (AS Border Router)

- Identified by a proper bit (*e* bit) in the router LSA

Backbone area

- All areas must be connected to it
- ALWAYS present

ABR (Area Border Router)

- It has the databases of the areas it belongs to
- Identified by a proper bit (*b* bit) in the router LSA

Multi-area OSPF: configuration

For configure area border router

- In the ABR, the association between interfaces and areas must be specified:
  - **R(config)# interface X Y/Z**
  - **R(config-if)# ip ospf *OSPF_ID* area *areaID***
- The network of different areas will be identified in the routing table with the following acronym: ***O IA***

Antonio Cianfrani

Dynamic Host Configuration Protocol

(DHCP)

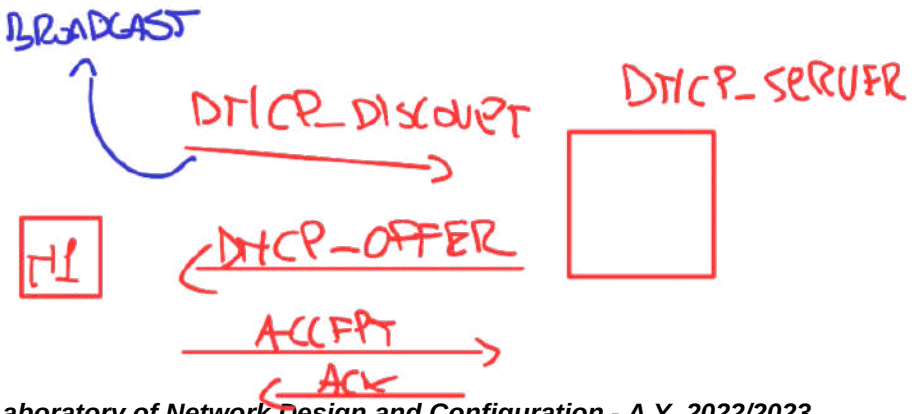
DHCP protocol

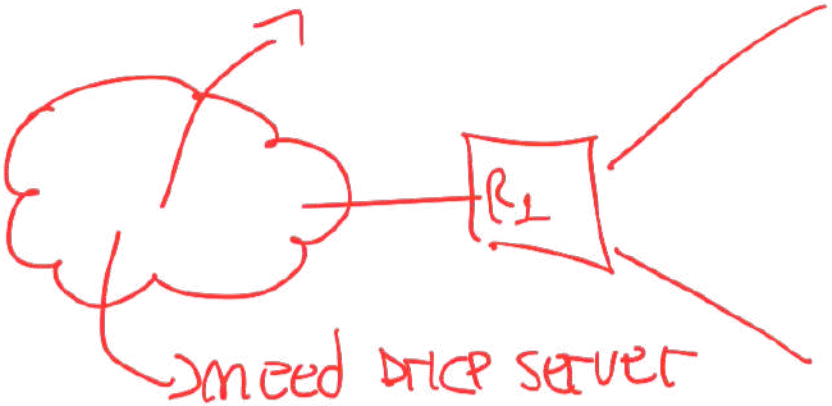
in actual network DHCP_SERVER is integrated in ROUTER

- The DHCP protocol allows to dynamically configure hosts in a LAN.
- Configuration parameters: for hosts
  - ✓ <u>IP address and subnet mask</u>
  - ✓ Default gateway
  - ✓ DNS Server

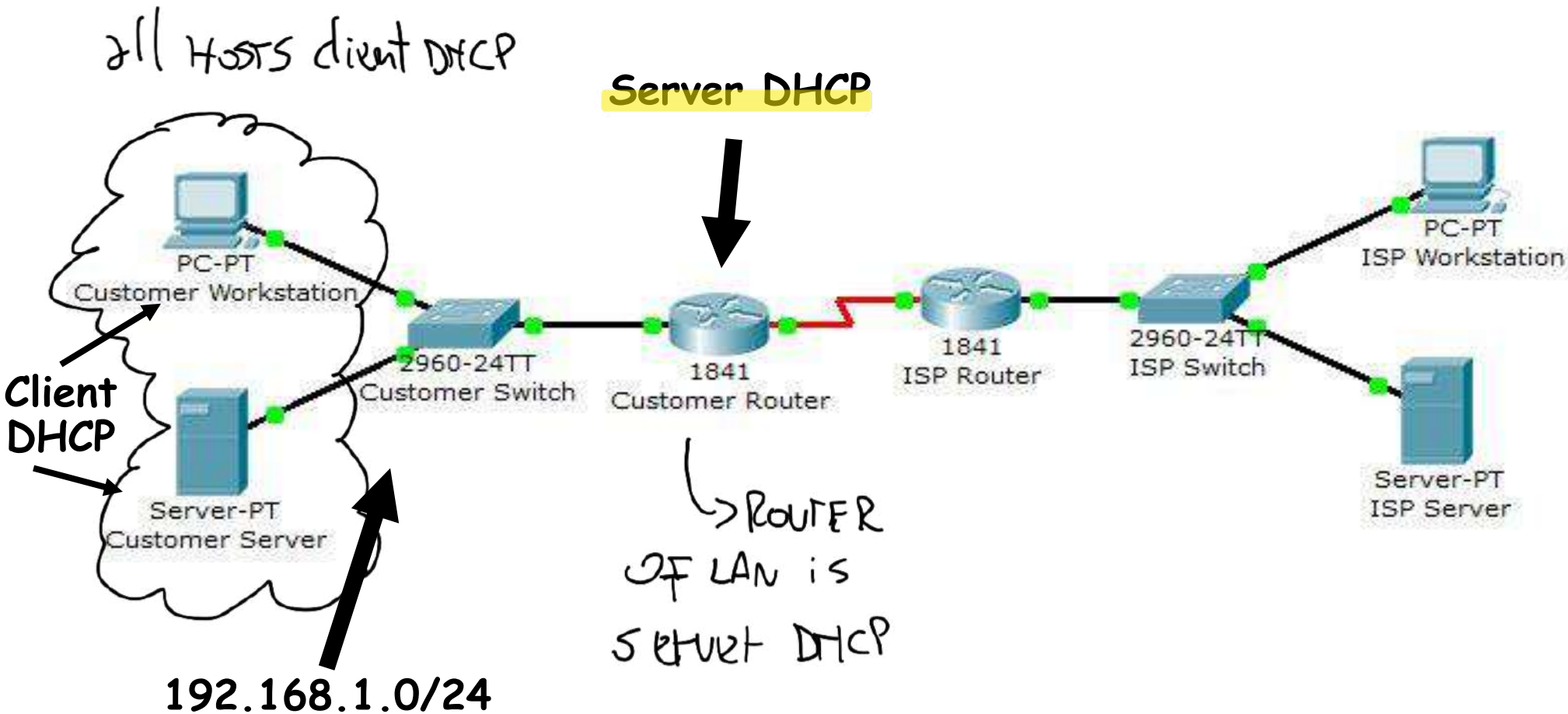
host dynamically configuring

DHCP RUN HERE





Example


all HOSTS client DHCP
Server DHCP
PC-PT
Customer Workstation
Client DHCP
Server-PT
Customer Server
192.168.1.0/24
2960-24TT
Customer Switch
1841
Customer Router
ROUTER OF LAN is SERVER DHCP
1841
ISP Router
2960-24TT
ISP Switch
PC-PT
ISP Workstation
Server-PT
ISP Server

Server DHCP configuration (1/3)

- The command to start DHCP configuration is

```
Router(config)#ip dhcp pool NAME_POOL
```

- NAME_POOL is the name used to identify the DHCP configuration on the router
- The next step is the definition of the address pool to be assigned dynamically:

```
Router(config-dhcp)# network NET_ADDRESS NETMASK
```

- In the case of 192.168.1.0/24 block:

```
Router(config-dhcp)# network 192.168.1.0
255.255.255.0
```

Server DHCP configuration (2/3)

- It could happen that the set of available IP addresses to be assigned dynamically is not exactly the pool
- Motivation: a host (usually a server) requires a static (fixed) IP address
- It is possible to exclude IP address from the pool

```
Router(config)#ip dhcp excluded-address IPADDRESS(ES)
```

- If the initial 49 addresses should be excluded:

```
Router(config)#ip dhcp excluded-address 192.168.1.1
                                        192.168.1.49
```

remove from pool addresses assigned statically

Server DHCP configuration (3/3)

- To provide the Internet access, default router (192.168.1.1) and DNS server (192.168.1.10) must be configured

now DHCP configuration

```
Router(config-dhcp)# default-router 192.168.1.1
```

```
Router(config-dhcp)# dns-server 192.168.1.10
```

Antonio Cianfrani

Network Address Translation

(NAT)

Public and private IP addresses

- Problem: IP addresses are not enough to assign a unique IP address to each network host/device

must use public addresses for communicate over internet

Address Class	Number of Network Numbers Reserved	Network Addresses
A	1	10.0.0.0
B	16 blocks	172.16.0.0 - 172.31.0.0 /16
C	256	192.168.0.0 - 192.168.255.0 /24

this are private IP addresses, can not be used in public internet

router must implement a translation mechanism

NAT (1/4)

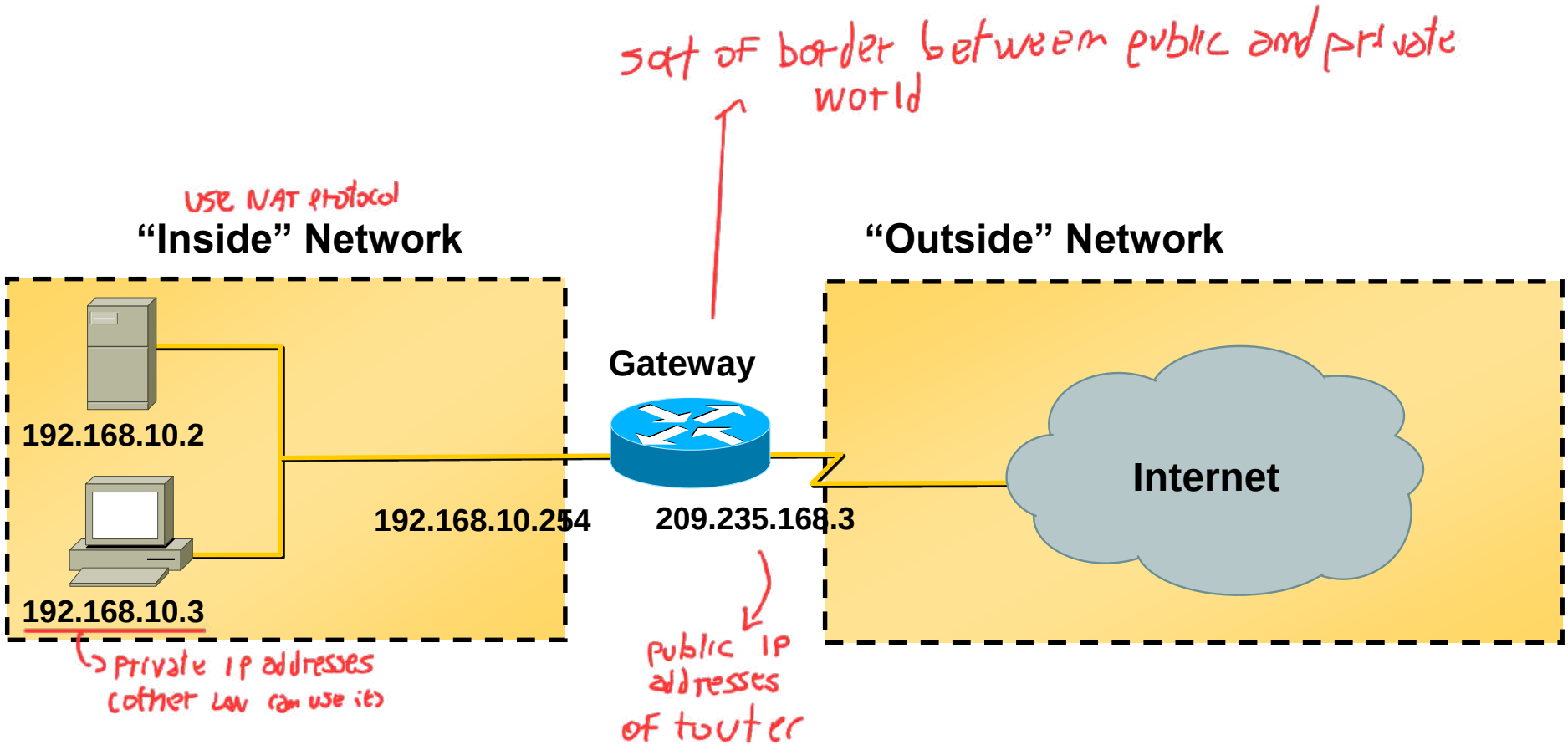
Network address translation

↳ we break separation of layer for implement it (IP/TCP)

- **Advantages**
  - No need to request public IP addresses for all the devices
  - Reduction of public IP addresses
  - Security improvement: real devices IP addresses hidden
- **Drawbacks:**
  - Extra-load for access routers
  - Breaking of the layers separation rule

in different LAN you can use same private IP addresses

NAT (2/4)


set of border between public and private world
use NAT protocol
"Inside" Network
"Outside" Network
Gateway
192.168.10.2
192.168.10.3
192.168.10.254
209.235.168.3
Internet
private IP addresses (other LAN can use it)
public IP addresses of router

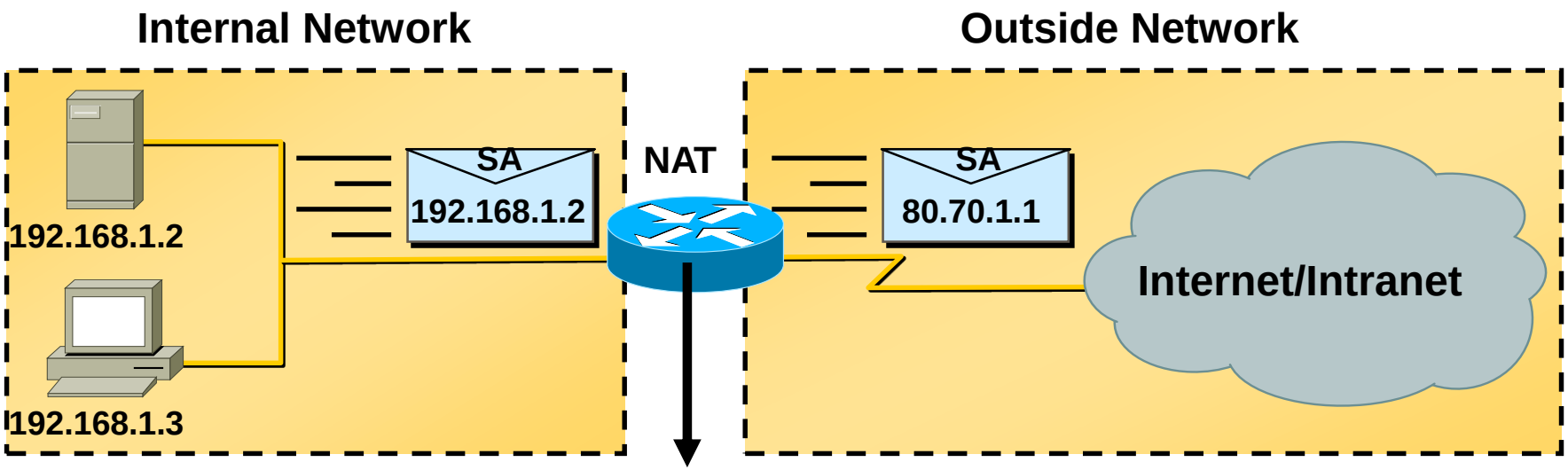
NAT (3/4)

- The IP addresses of the local network (192.168.10.X) are not "visible" from outside.
- The router (Gateway) has two network interfaces, one on the local (inside) network and one on the public (outside) network.
- The router is connected to the Internet Service Provider (ISP), that will assign the public IP address to the outside interface

NAT (4/4)

- The router must implement NAT (Network Address Translator - RFC 1631)
- The NAT idea id to modify (translate) the IP addresses (and other header fields) of outgoing and incoming packets.
- Two main types of NAT operations exists: (three in reality)
  - Static NAT: one-to-one translation;
  - Dynamic NAT: N-to-one translation.

Static NAT



NAT Table

Inside Local IP Address	Inside Global IP Address
192.168.1.2	80.70.1.1
192.168.1.3	80.70.1.2

1 to 1 association

associate one public ip address one local ip address

The translation is performed also for the incoming packets (acting on the IP destination address field)

Dynamic NAT

exploit source port number for differentiate different packet use same public address but use different source port

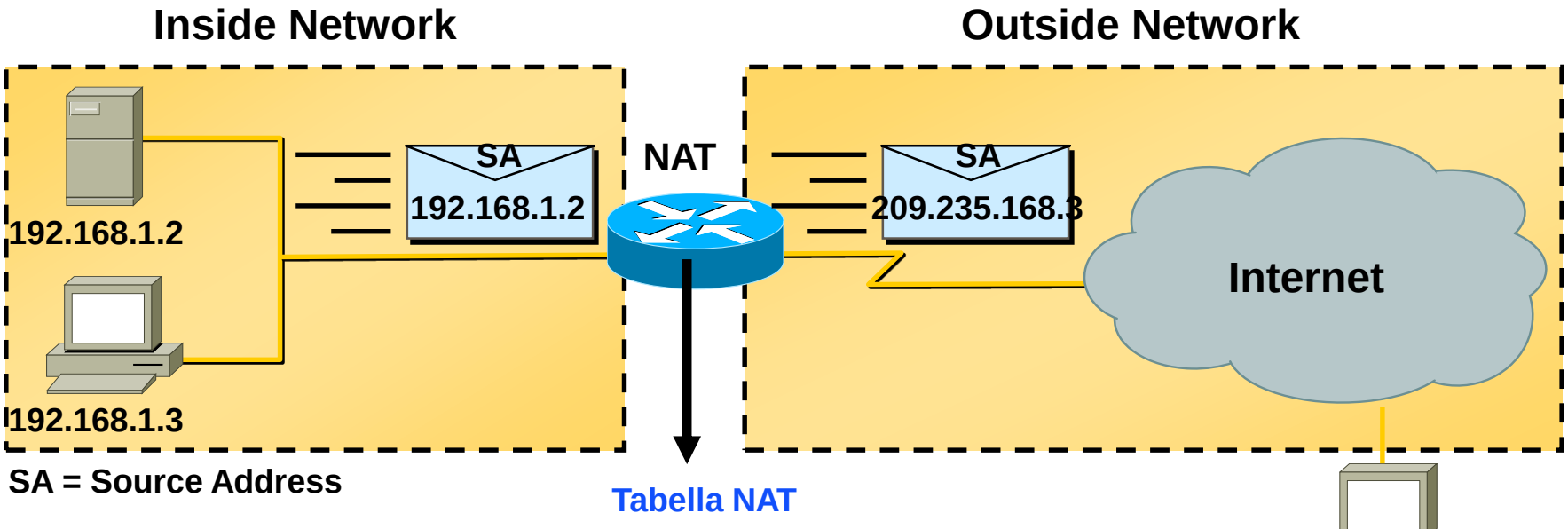


Tabella NAT

Inside Local IP Address & port	Inside Global IP Address & port
private ip address ↑ spoft number ↑	
192.168.1.2:1024	209.235.168.3:55001
192.168.1.3:1723	209.235.168.3:55002

router must work also on transport layer, break separation of layers.

NAT Configuration

- The first step in NAT configuration is the setting of inside (local) and outside (global) interfaces

```
Router(config-if)#ip nat inside

Router(config-if)#ip nat outside
```

Static NAT configuration

- One-to-one translation among a private and a public IP address.
- Used for servers.
- Translation rule:

```
Router(config)# ip nat inside source static
                private_address public_address
```

Dynamic NAT configuration: pool of IP addresses

- The number of available public IP addresses is lower than the number of hosts (private IP addresses) requiring Internet access

1-99

- Private IP addresses definition:

```
Router(config)# access-list access-list-number
permit source address wildcard
```

LAN

- Pool of public IP addresses (carefully chosen!!!!):

```
Router(config)# ip nat pool name start-ip end-ip
netmask netmask
```

.131 → .133

- Translation rule

```
Router(config)# ip nat inside source list acl-
number pool name
```

access-list-num

Dynamic NAT configuration: overload

only use public address of interface of router

- It is also possible to use a single public IP address for all the LAN hosts: the public IP address of the router
- In this case the pool of public IP addresses is not required
- Translation rule:

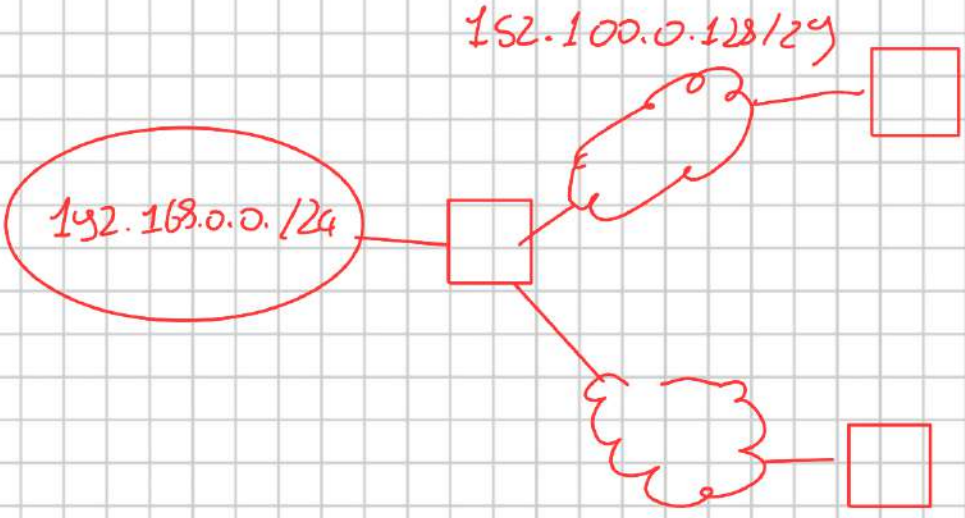
```
Router(config)# ip nat inside source list acl-number interface interface overload
```

numbers

o private ip configuration also required

show ip nat translation
show running-config

10.0.0.0 private ip-addresses



[.131 → .133]
Pool

two /30 used → no pool - no aviable public IP address
in /30 only two addresses used by interfaces of router

For IP: /26 32-26 = 6 $2^6 = 64$ 256-64 = 192 /26 = 255.255.255.192 → wildcard 0.0.0.63

1) interface Fa0/0
a. → IP address a.b.c.d. A.B.C.D.
clock rate 4
no shutdown

Given the network topology reported in the Figure, the student has to configure the following devices: Router0, PC0, PC1and Server0. The following operations must be executed.

1) IP addresses assignment strategy is based on the following guidelines:

   a. Router0 interfaces must use the last available address of the addresses block (in the case of clock, use 64000 value).

   b. IP addresses for network 151.0.0.0/28 must be assigned statically (IP address 151.0.0.10 must not be used).

   c. IP addresses for network 10.8.0.0/26 must be assigned dynamically, excluding from the block of dynamic IP addresses the first 8 available addresses. Server0 must have IP address 10.8.0.23

2: a), b)

For router0, 3 public network and 10.8.0.0 is private, only enable on other 3 ospf:
- router ospf 1
- network 151.0.0.0 0.0.0.15 area 0
- network 123.0.0.8 0.0.0.3 area 0
- network 123.0.0.12 0.0.0.3 area 0
- passive-interface Fa 1/0

↓
to check: show ip route

2) Routing must be configured considering that:

   a. OSPF is already enabled on routers Router1, Router2 and Router3; → not able in private network!

   b. the number of control messages should be minimized;

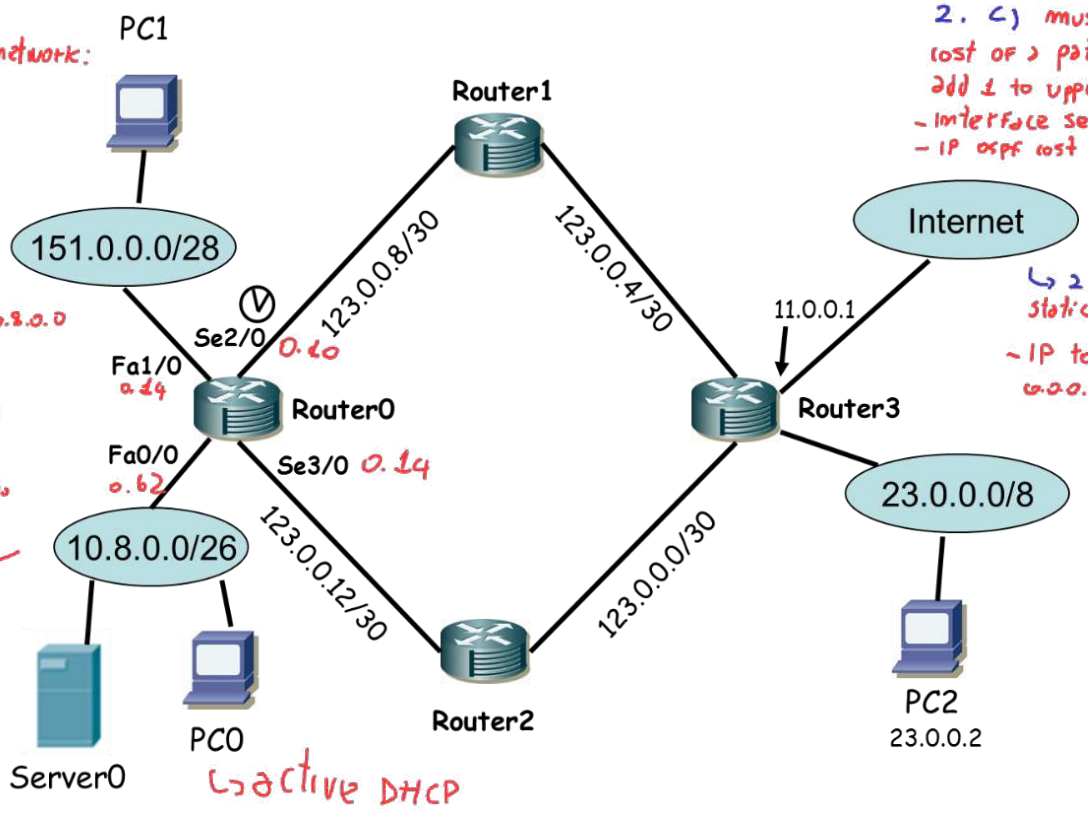
   c. the path to network 23.0.0.0/8 must be the one crossing Router2, while all other paths must be the shortest ones.

   d. The path to reach Internet must be the one crossing Router1.

3) NAT must allow hosts of private networks to share the IP address of interface Se2/0 of Router0; as an exception, Server0 has to use public IP 151.0.0.10.

3) NAT for private network:
- interface Fa 1/0
  se 2/0
  se 3/0
  ip nat outside
- interface Fa 0/0
  ip nat inside
- access-list permit 10.8.0.0 0.0.0.63
- ip nat inside source list 1 interface se2/0 overload
- ip nat inside source static 10.8.0.23 151.0.0.10

Check Ping PC0 to PC2 and viceversa



↳ active DHCP

2. c) must change cost of a path for example add 1 to upper path:
- interface se2/0
- IP ospf cost 65
↳ default 64 + 1

↳ 2. d) must add static route:
- IP route 0.0.0.0 0.0.0.0 123.0.0.4
↳ next hop

C.

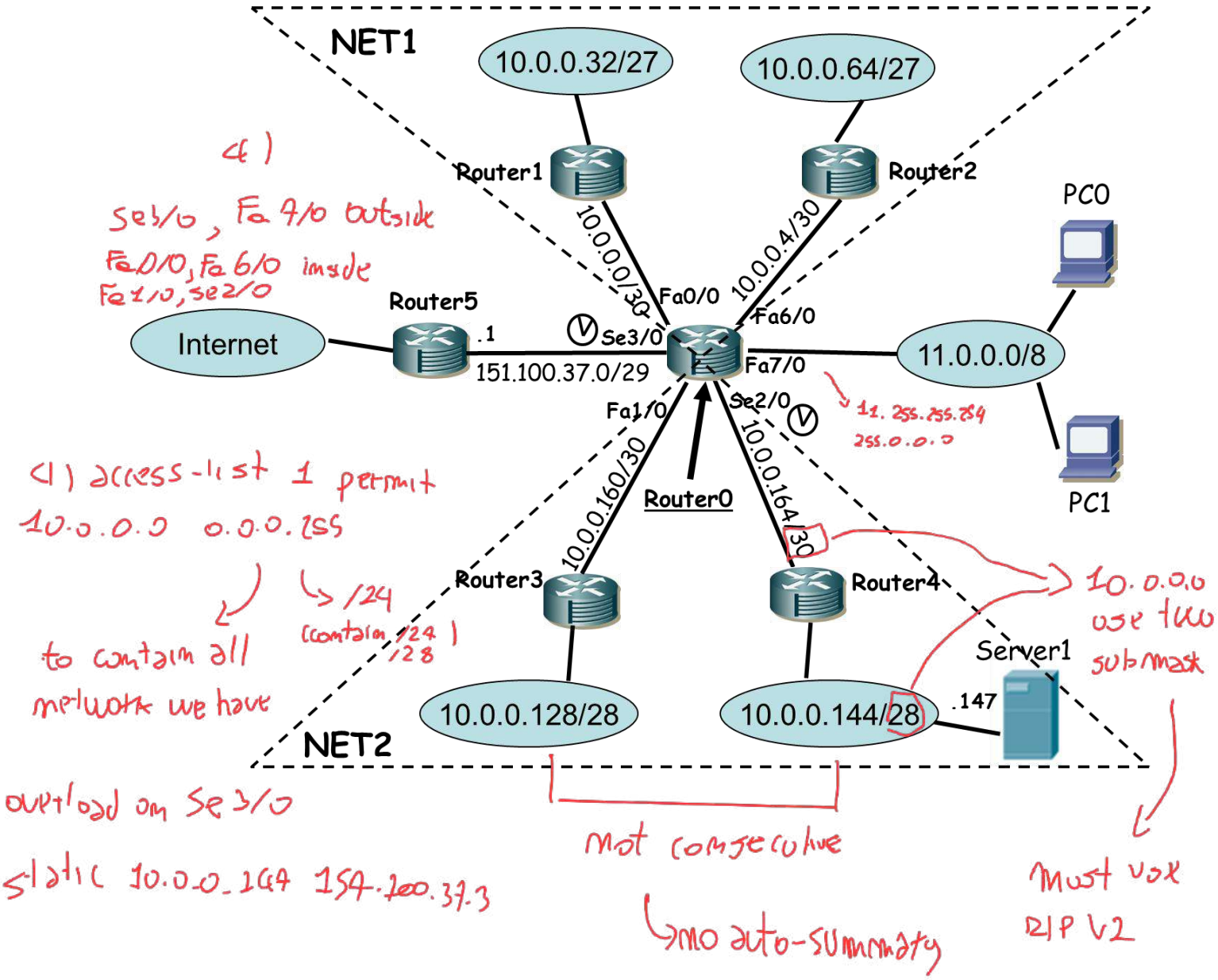
Dynamically:
IP dhcp pool PROVA
network a.b.c.d A.B.C.D.
default router a.b.c.d
exit

IP dhcp excluded address IP-of-server0
0.1 to 0.8

32-8 = 24

The logical topology of the network to configure is reported in the figure. The devices to be configured are Router0, PC0 and PC1.

1) The IP addresses should be assigned considering the following rules.
   a. The interfaces of Router0 must have the last IP address of the blocks (in the case of a clock rate, value 64000 should be used).
   b. In the network 11.0.0.0/8, addresses must be assigned dynamically; the first useful 10 addresses must be excluded by the dynamic assignment procedure.
2) OSPF protocol must be enabled on NET1 network portion, while RIP protocol must be enabled on NET2 network portion.
3) All networks must have an Internet access configured. → static route 0.0.0.0 0.0.0.0 rip and ospf distribute origin
4) NAT must be implemented so that all hosts will use the public IP address of Se3/0 interface of Router0; the only exception is Server1 bthat must use an available IP address in the block of network between Router0 and Router5.



3) access-list 1 permit 10.0.0.0 0.1.255.255 → /15 for contain 10.0.0.0 10.1.0.0

- Fa0/0, Fa1/0 inside
- Se2/0 3/0 6/0 outside

ip nat inside source list 1 interface Se6/0

ip nat inside source static 10.1.0.18 12.0.0.10

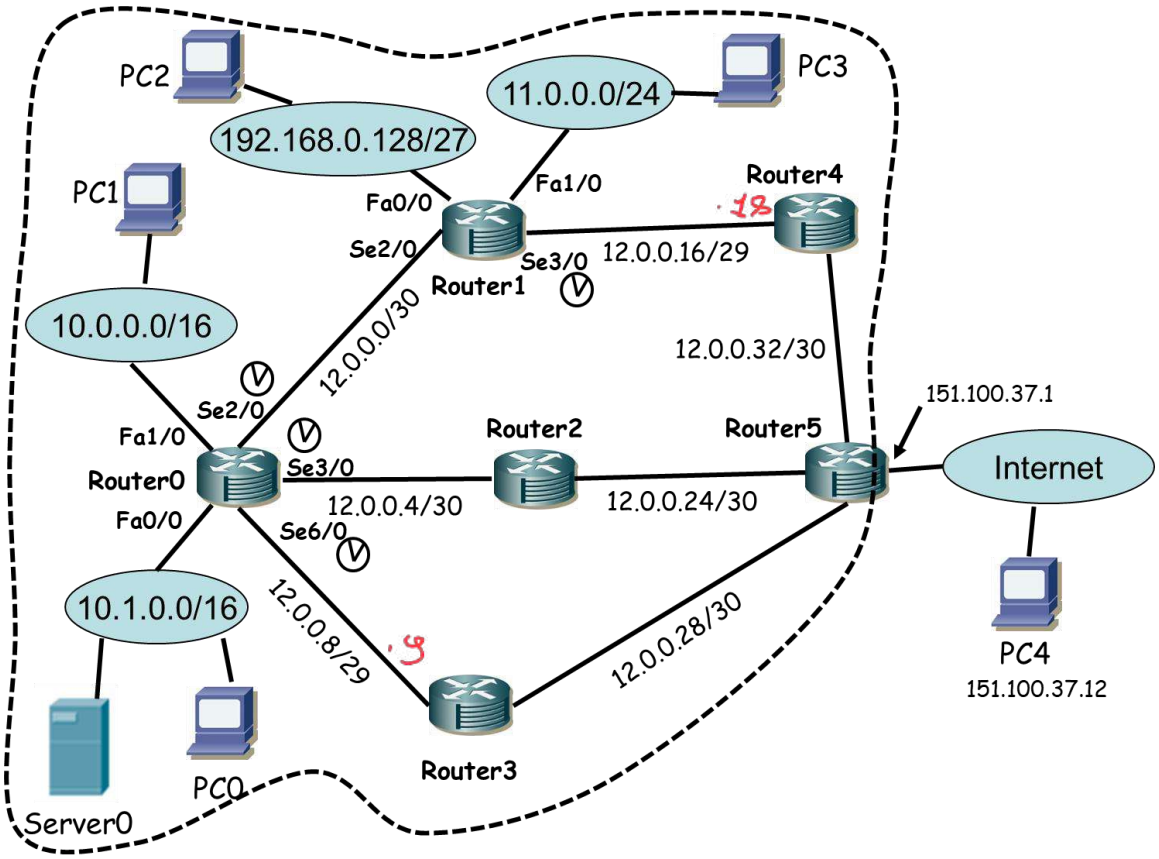
Given the network topology reported in the Figure, the student has to configure the following devices: Router0, Router1, PC0, PC1, PC2, PC3 and Server0. The dotted line represent the area managed by the same administrative entity (ISP). The following operations must be executed.

1) IP addresses assignment strategy is based on the following guidelines:
   a. Router0 interfaces must use the last available address of the addresses block (in the case of clock, use 64000 value).
   b. Router1 interfaces must use the first available address of the addresses block (in the case of clock, use 64000 value).
   c. IP addresses for networks 10.0.0.0/16, 192.168.0.128/27 and 11.0.0.0/24 must be assigned statically.
   d. IP addresses for network 10.1.0.0/16 must be assigned dynamically, excluding from the block of dynamic IP addresses the first 15 available addresses. Server0 must have IP address 10.1.0.18

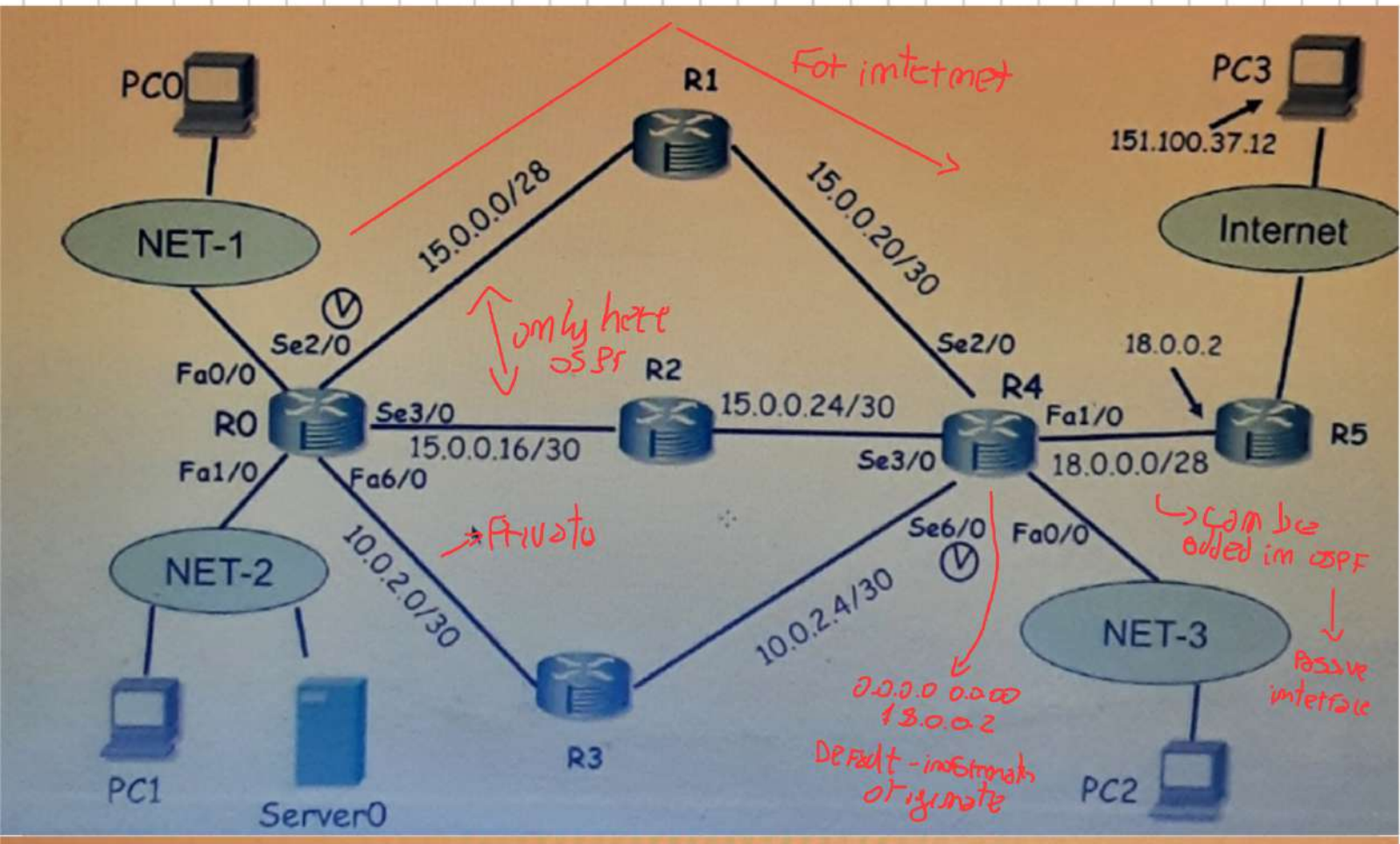
   No add private networks → 10.0.0.0, 10.1.0.0, 192.168.0.128

2) Routing must be configured considering that:
   a. OSPF is already enabled on Router2, Router3, Router4 and Router5, and links costs are 1 for FastEthernet interfaces and 64 for Serial interfaces;
   b. the number of control messages should be minimized; → passive-int on Fa1/0 (r1)
   c. for Router0, OSPF must not use Router3 as next hop while Internet must be reached using Router3 as next hop; → increase Se6/0 to cost 129
   d. for Router1 Internet must be reached using Router4 as next hop. → 0.0.0.0 0.0.0.0 next hop

3) NAT must be configured where needed using a feasible solution (your choice).



MIDTERM 2021/2022



Given the network topology reported in the Figure, the student has to configure the following devices: R0, R4, PC0, PC1, PC2 and Server0. The following operations must be executed.

1) IP addresses assignment strategy is based on the following guidelines:
   a. R0 and R4 interfaces must use the first available address of the addresses block (in the case of clock, use 64000 value).
   b. NET-1: 15.0.0.128/25 , NET-2: 10.0.1.0/24 , NET-3: 10.1.0.0/24
   c. IP addresses for networks NET-1 and NET-3 must be assigned statically.
   d. IP addresses for network NET-2 must be assigned dynamically (Server0 -> 10.0.1.22)
2) Routing must be configured considering that: OSPF only on public area, non in NET-2-3
   a. OSPF must be enabled among routers R0, R1, R2 and R4 (not R3 and R5!); links costs are 1 for FastEthernet interfaces and 64 for Serial interfaces;
   b. OSPF must be used for the default path (Internet) and router R0 must reach Internet using R1 as next-hop; decrease cost of R0 → R1
   c. the number of OSPF control messages should be minimized;
   d. for R0, the path having R3 as next-hop must be configured as backup path to reach Internet. add a static route for 0.0.0.0 with R3 next hop with greater AD
3) NET-1, NET-2 and NET-3 must have Internet access. NAT configuration must be executed using a pool of public IP addresses.

Focus on switches now!

Antonio Cianfrani

Virtual LAN (VLAN)

switch(config)# line vty 0 15

-line)# password cisco ⇐ telnet

login

exit

enable password cisco ⇐ enable

Ethernet

- Ethernet standard (IEEE 802.3) for Local Area Network (LAN).
- CSMA/CD (Carrier Sense Multiple Access/Collision Detection)
- **Carrier Sense**: all devices have to "sense" the medium before sending frames.
- **Multiple Access**: collisions
- **Collision Detection**: Back-off algorithm

· main feature of Ethernet is multiple access control, how to allow access to a physical medium to different hosts.
· allow to be aware of collission and manage retrasmission

·IP addresses change if we change network, instead MAC is a fixed identifier

Ethernet frame

→ MAC Addresses

Preamble + Start of frame	Destination Address	Source Address	Lenght or Type	Layer 3 PDU (Ipv4 packet)	Frame Check Sequence

- MAC address: 6 bytes. Hexadecimal format: 00-24-9A-3C-74-02, 00:24:9A:3C:74:02, 0024.9A3C.7402

MAC 6 bytes — IP bytes

↳ use Hexadecimal Format (0-9, A-F)

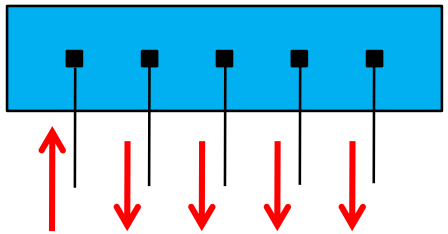
10+6 = 16 values

Ethernet devices

layer 1 device, is a repeater

- **Hub**: forwards frames on all interfaces

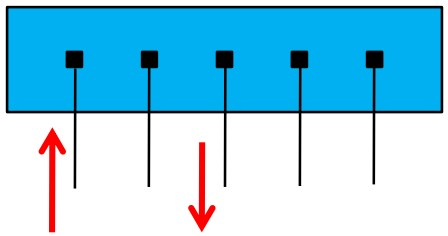
receive a bit on an interface, forward to all other



Broadcast

layer 2 device

- **Switch**: forwards frames only on the proper outgoing interface (the one to reach the destination). It has a MAC forwarding table.



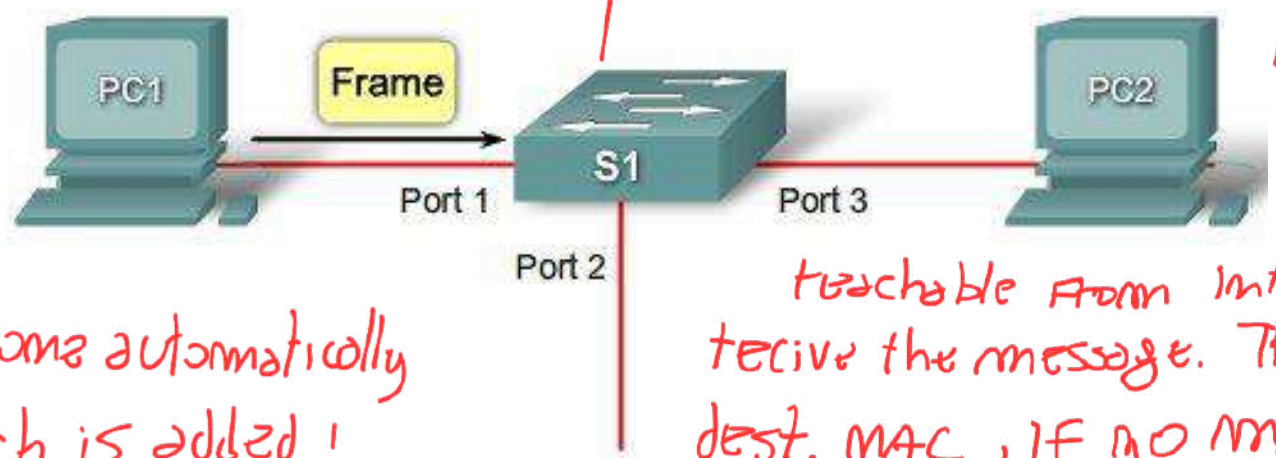
use a MAC table

MAC forwarding table

↳ computed by a learning procedure

- For each (known) MAC address, the proper outgoing interface.
- The MAC forwarding table is populated by means of a Mac learning procedure.

tecive Frame for PC1, read source MAC address and add it (PC1) in MAC forwarding table, that PC1 is reachable from interface that tecive the message. Then read dest. MAC, if no match in table, forward message to all interfaces.



this is done automatically when switch is added!

Basic switch configuration

only for management aspect.

- The switch IOS is similar to the router one
- The basic configuration parameters are:
  - IP address and subnet mask (?)
  - Default gateway (?)

can assign them to a switch: for do a configuration of a switch if we are not physicall connected, only way is to use a telnet connection.

only IP switch need to exchange data with something outside ethernet connection

- Only for switch management: **Telnet connection**

to get access to CLI of a switch

  - Passwords must be configured to provide access and configuration

DEFAULT GATEWAY:

- The default gateway needs to provide access to an external web/tftp server (mainly to download a new IOS version)

update IOS version of switch

Management interface

- we assign IP address not to an interface, but to the all switch.
- we assign IP address to a virtual interface 'vlan 99'.

Cisco IOS CLI Command Syntax	
Switch from privileged EXEC mode to global configuration mode.	S1#**configure terminal**
Enter the interface configuration mode for the VLAN 99 interface.	S1(config)#**interface vlan 99** . virtual interface -all switch
Configure the interface IP address.	S1(config-if)#**ip address 172.17.99.11 255.255.255.0**
Enable the interface. (virtual)	S1(config-if)#**no shutdown**
Return to privileged EXEC mode.	S1(config-if)#**end**
Enter global configuration mode.	S1#**configure terminal**
Enter the interface to assign the VLAN. specify on what real interface is possible to access the virtual one	S1(config)#**interface fastethernet 0/18** real interface
Define the VLAN membership mode for the port.	S1(config-if)#**switchport mode access**
Assign the port to a VLAN. (map real to virtual	S1(config-if)#**switchport acces vlan 99**
Return to privileged EXEC mode.	S1(config-if)#**end**
Save the running configuration to the switch start-up configuration.	S1#**copy running-config startup-config**

Virtual - LAN

VLAN: motivations

(internet networks) → physical infrastructure → need a block of IP addresses

- LAN Ethernet: all hosts connected to the same physical infrastructure (switches, hubs, cavi) belong to the same IP network.

(separated networks → same pysical networks)

- If several IP networks are needed in the same physical location (i.e. a building), several physical infrastructures are required.

- The introduction of Virtual LAN (VLAN) feature allows for the sharing of the same physical infrastructure among different IP networks.

- VLAN: IP network sharing a physical network with different VLANs.

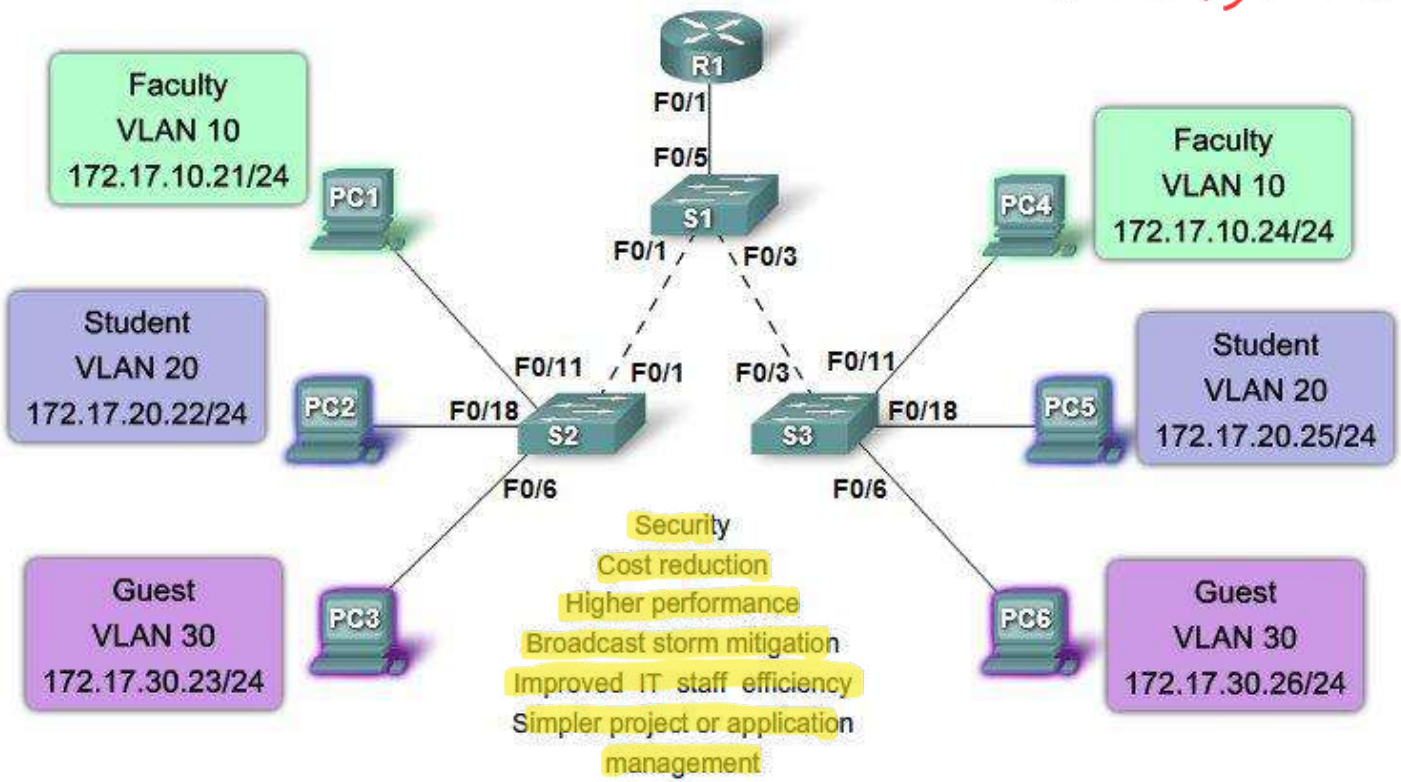
VLAN advantages

- Security
- Cost saving
- Reduction of broadcast traffic

instead of replicating the same physical network 3 times, we use 3 logical networks

· PC1 - PC2 same physical network, but different logical network.

· PC1 can send frame only to PC4, same logical network



three IP networks (logical)
↓
using three switches with same router

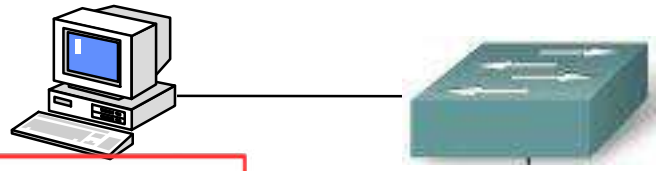
VLAN: configuration

- Switches must be properly configured.
- Each VLAN is identified by a number (VLAN ID: 1 - 1005) and has its own address block
- The VLANs of a network must be defined in all the switches:

Switch (config)# vlan x no constraint

Switch (config-vlan)# name *name* (optional)

Ex: 3 net
- vlan 1
- vlan 2
- vlan 3



VLAN 10
"student"

x name

Example:

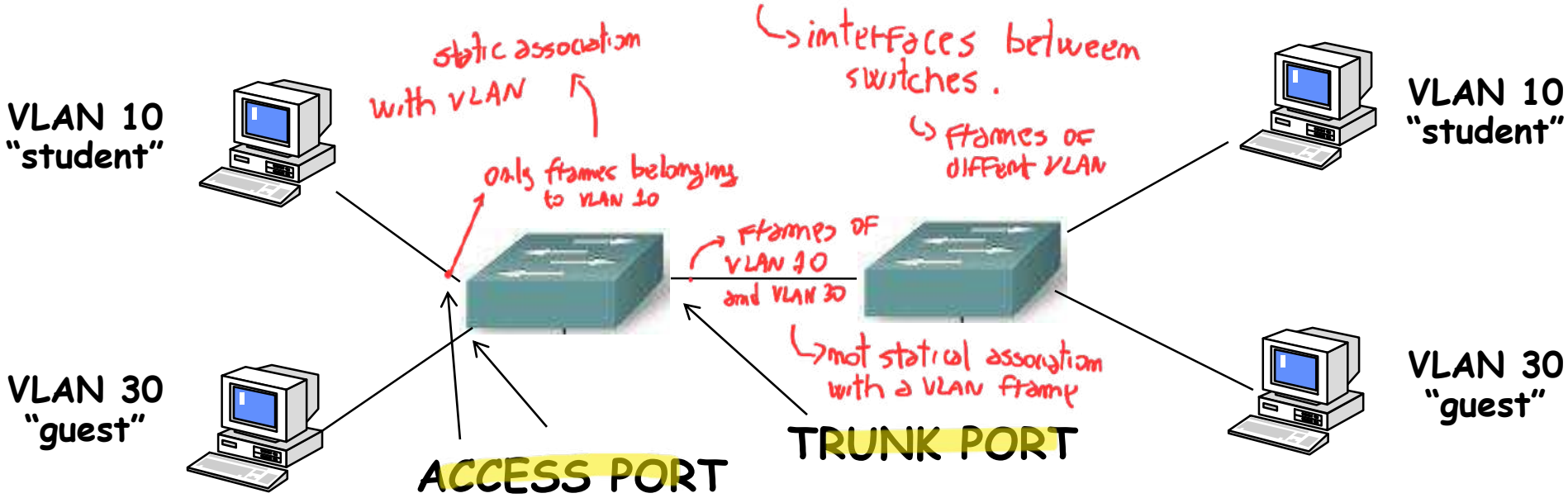
Switch (config)# vlan 10 needed

Switch (config-vlan)# name student

if we want, usefull to remember for us.

Switch ports

- **Access port**: connected to hosts belonging to a single VLAN → interfaces directly connected to hosts, only frames belonging to a single VLAN
- **Trunk port**: receiving and forwarding frames belonging to different VLANs → a mechanism to differentiate frames of different VLANs is needed



Access Port

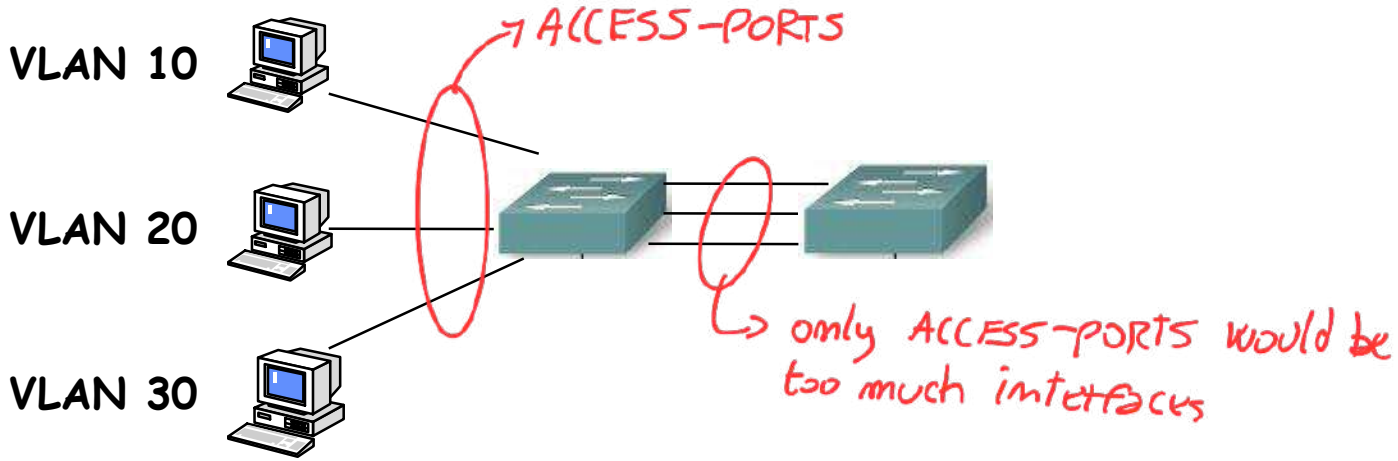
Fa 1/0 $\xrightarrow[\text{statically connected}]{}$ VLAN X

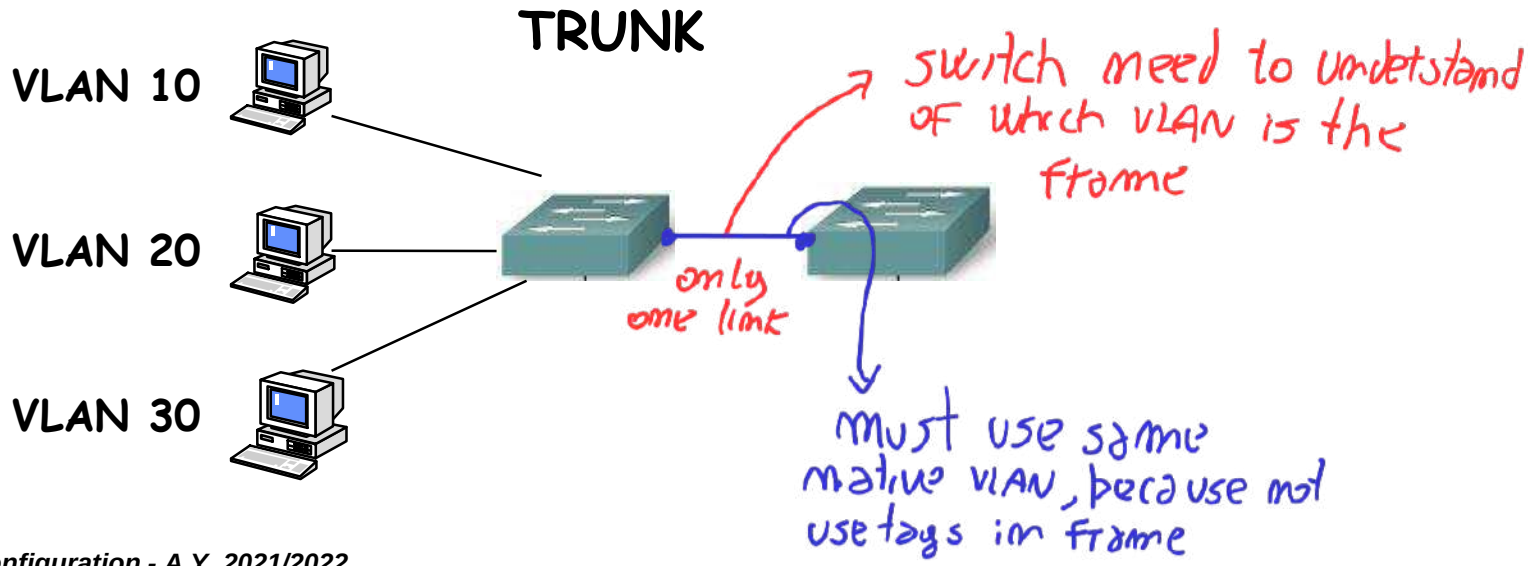
- Port-based VLAN → access port are statically associated to a specific VLAN.
  → only send/receive frames of VLAN X
- Configuration:

```
Switch (config)# interface FastEthernet 1/0
Switch (config-if)# switchport mode access
Switch (config-if)# switchport access vlan X
```

→ Configure an Access port, statically

Access ports limitation

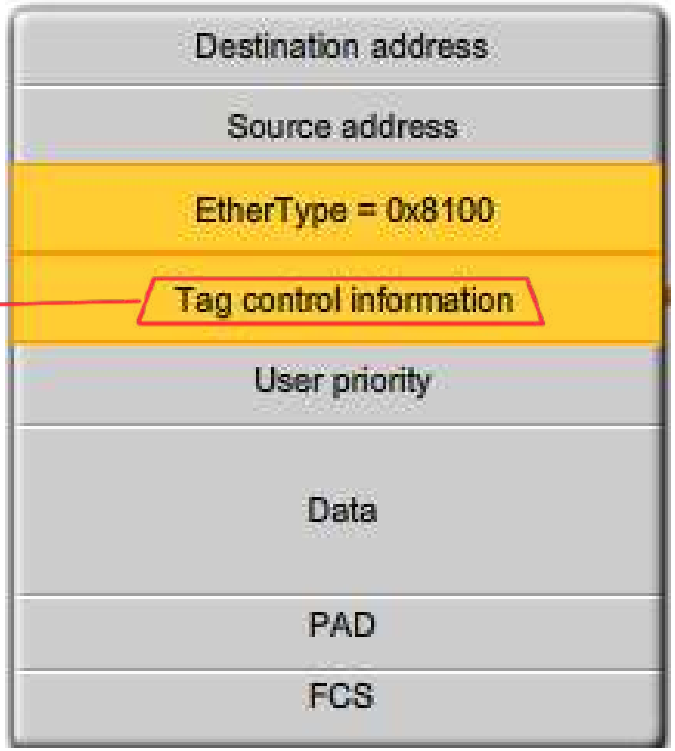




Trunk Port (1/3)

- How to associate a VLAN to a received frame on a trunk port?
- Ethernet standard must be extended.
- 802.3Q: a tag is inserted in the ethernet header to insert the VLAN ID → "tagged" frames

Ethernet extended



specify VLAN identifier

For default: no native VLAN, will use 1 for native

Trunk Port (2/3)

- With the ***trunk*** word, we refer to the point-to-point link among directly connected trunk ports
- The frames crossing a trunk are all "tagged". The only exception regards frames belonging to a single VLAN: the Native VLAN.

VLAN_ID

```
Switch (config)# interface FastEthernet o/x
Switch (config-if)# switchport mode trunk
Switch (config-if)# switchport trunk native vlan 99
```

trunk port

only one mandatory

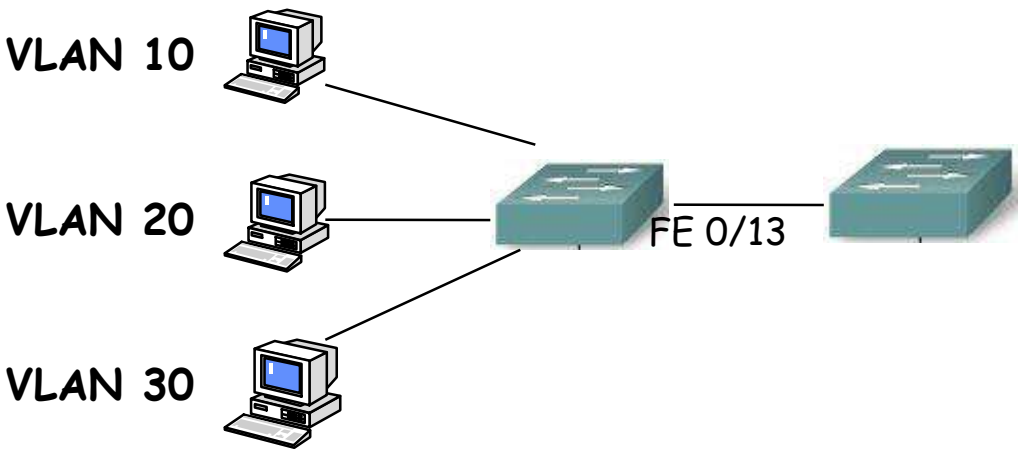
optional

native VLAN not need tag, reduce overhead

Trunk Port (3/3)

- By default the trunk port forward and receive all VLANs frames.
- It is also possible to configure the trunk port so that only a subset of VLANs is allowed:

Switch (config-if)# switchport trunk allowed vlan y

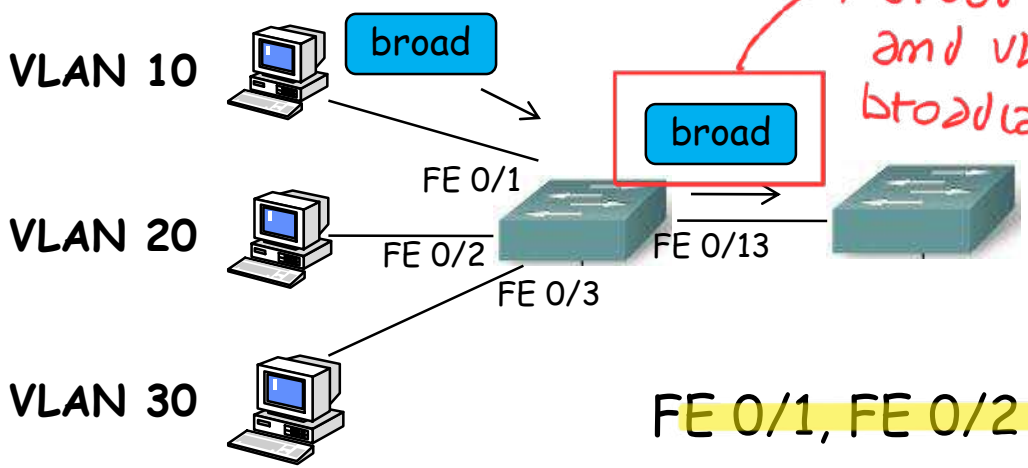



```
R(config)# interface FastEthernet 0/13
R(config-if)# switchport mode trunk
R(config-if)# switchport trunk native vlan 99
R(config-if)# switchport trunk allowed 10-20
```

30 not allowed

Switch functioning

- Switch forwarding operations depends on the VLANs configuration.
- The broadcast traffic received on an access interface will be forwarded only on the interfaces "connected" to the same VLAN of the access port.



broadcast on trunck, VLAN 20 and VLAN 30 are different VLAN, broadcast only to same VLAN

↳ reduce number of broadcast messages

FE 0/1, FE 0/2 e FE 0/3: access port

FE 0/13: trunk port (allowed 10,20,30)

VLAN management (1/2)

- The ***show vlan brief*** command lists all the VLANs configured and the associated access interfaces

```
S1#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/1, Fa0/2, Fa0/4, Fa0/5
                                                Fa0/6, Fa0/7, Fa0/8, Fa0/9
                                                Fa0/10, Fa0/11, Fa0/12, Fa0/13
                                                Fa0/14, Fa0/15, Fa0/16, Fa0/17
                                                Fa0/18, Fa0/19, Fa0/20, Fa0/21
                                                Fa0/22, Fa0/23, Fa0/24, Gi0/1
                                                Gi0/2
```

No VLANs configured

```
S1#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/1, Fa0/2, Fa0/3, Fa0/4
                                                Fa0/5, Fa0/6, Fa0/7, Fa0/8
                                                Fa0/9, Fa0/10, Fa0/11, Fa0/12
                                                Fa0/13, Fa0/14, Fa0/15, Fa0/16
                                                Fa0/17, Fa0/19, Fa0/20, Fa0/21
                                                Fa0/22, Fa0/23, Fa0/24, Gi0/1
                                                Gi0/2
20   student                          active    Fa0/18
```

access-interface for VLAN student

VLAN 20 configured and associated to FastEthernet 0/18

VLAN management (2/2)

- The ***show interface FastEthernet x/y switchport*** command is used to check the mode of an interface

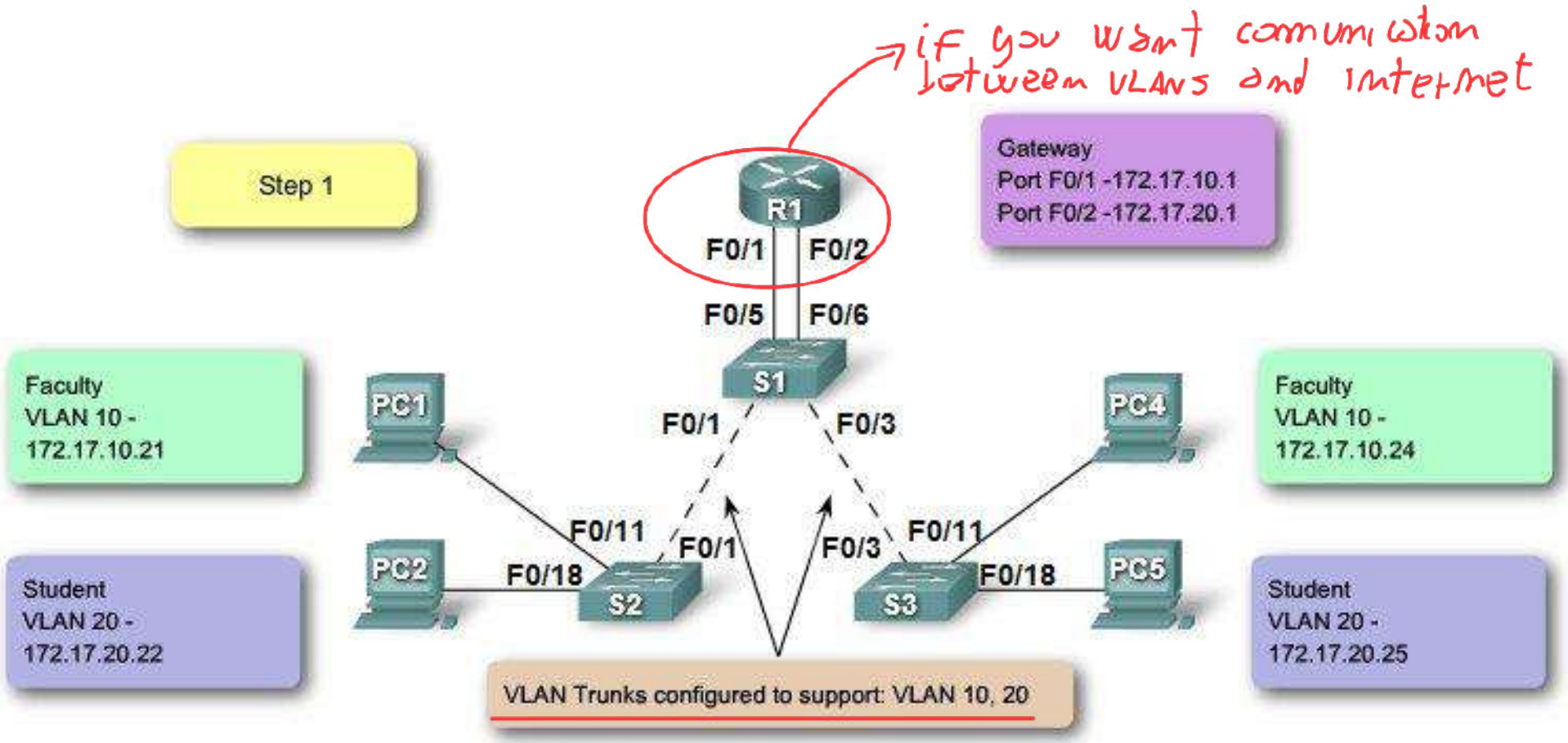
Access port

```
S1#show interfaces fa0/18 switchport
Name: Fa0/18
Switchport: Enabled
Administrative Mode: static access
Operational Mode: down
Administrative Trunking Encapsulation: dot1q
Negotiation of Trunking: Off
Access Mode VLAN: 20 (student)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk private VLANs: none
Operational private-vlan: none
```

Trunk port

```
S1#show interfaces f0/1 switchport
Name: Fa0/1
Switchport: Enabled
Administrative Mode: trunk
Operational Mode: down
Administrative Trunking Encapsulation: dot1q
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 99 (management)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: none
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk private VLANs: none
Operational private-vlan: none
Trunking VLANs Enabled: 10,20,30
```

Inter-VLAN routing



VLAN Troubleshooting

Misconfiguration of VLANs:

- Native VLAN mismatch
- Trunk mode mismatch
- Wrong IP addresses for VLAN hosts
- VLAN not allowed on trunk ports

trunk link : same native VLAN

TRUNK TRUNK

↳ not can ACCESS

Management VLAN where ip of switches are taken, VLAN 99 must be used to take IP of switches.

VLAN 99 (MANAGEMENT) ⟶ IP OF SWITCHES

Antonio Cianfrani

Inter-VLAN Routing

Inter-VLAN Routing

- In a LAN with many VLANs, the communication among host belonging to different VLANs (inter-VLAN routing) is not possible.
- To allow inter-VLAN routing, a router must be present.
- Two different ways to implement inter-VLAN routing:
  - ✓ Traditional Inter-VLAN;
  - ✓ "Router-on-a-stick" Inter-VLAN.

Traditional Inter-VLAN

- A router must be connected to a switch.
- The router must be connected to the switch with a certain amount of physical interfaces.

10 VLAN: Router to 10 physical interfaces

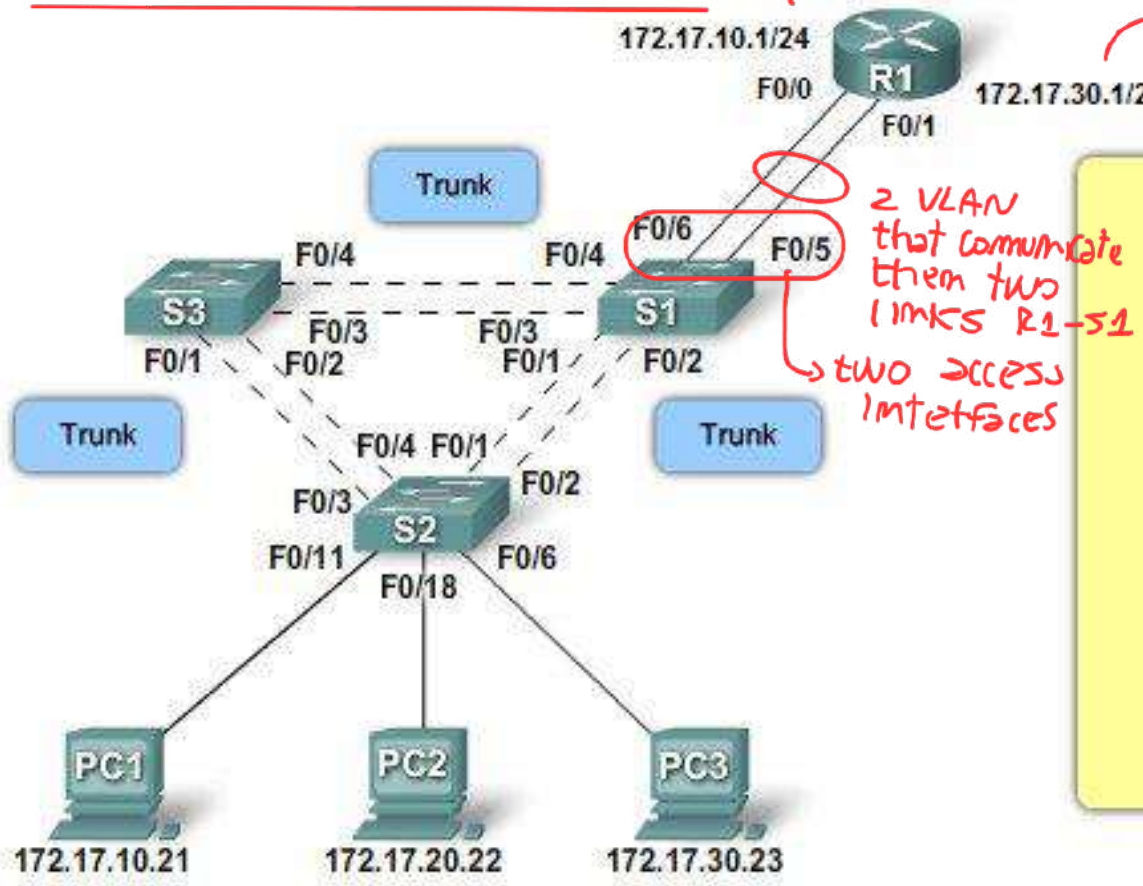
- The number of router-to-switch physical links is equal to the number of VLANs able to communicate each other.
- Each router interface is associated to a VLAN → an IP address of the VLAN block must be assigned to it.
- The switch ports connected to the router must be configured in **access** mode.

Traditional Inter-VLAN: example (1/3)

- Assumption: only VLAN 10 and VLAN 30 are allowed to communicate each other.

host of VLAN 10

host of VLAN 30



2 VLAN that communicate them two links R1-S1

two access interfaces

Switch S1 Ports

F0/6 = VLAN10

F0/5 = VLAN30

F0/1-F0/4 = Trunk

Switch S2 Ports

F0/11 = VLAN10

F0/18 = VLAN 20

F0/6 = VLAN30

F0/1-F0/4 = Trunk

Traditional Inter-VLAN: example (2/3)

assiym IP address

same as always

- **Router**: interfaces configuration

```
R1(config)# interface Fa 0/0
R1(config-if)#ip address 172.17.10.1  255.255.255.0
R1(config-if)# no shutdown

R1(config)# interface Fa 0/1
R1(config-if)#ip address 172.17.30.1  255.255.255.0
R1(config-if)# no shutdown
```

IP addresses of VLAN host

Traditional Inter-VLAN: example (3/3)

- **Switch**: configuration of the ports connected to the router

```
S1(config)# interface Fa 0/6
S1(config-if)# switchport access vlan 10

S1(config)# interface Fa 0/5
S1(config-if)# switchport access vlan 30
```

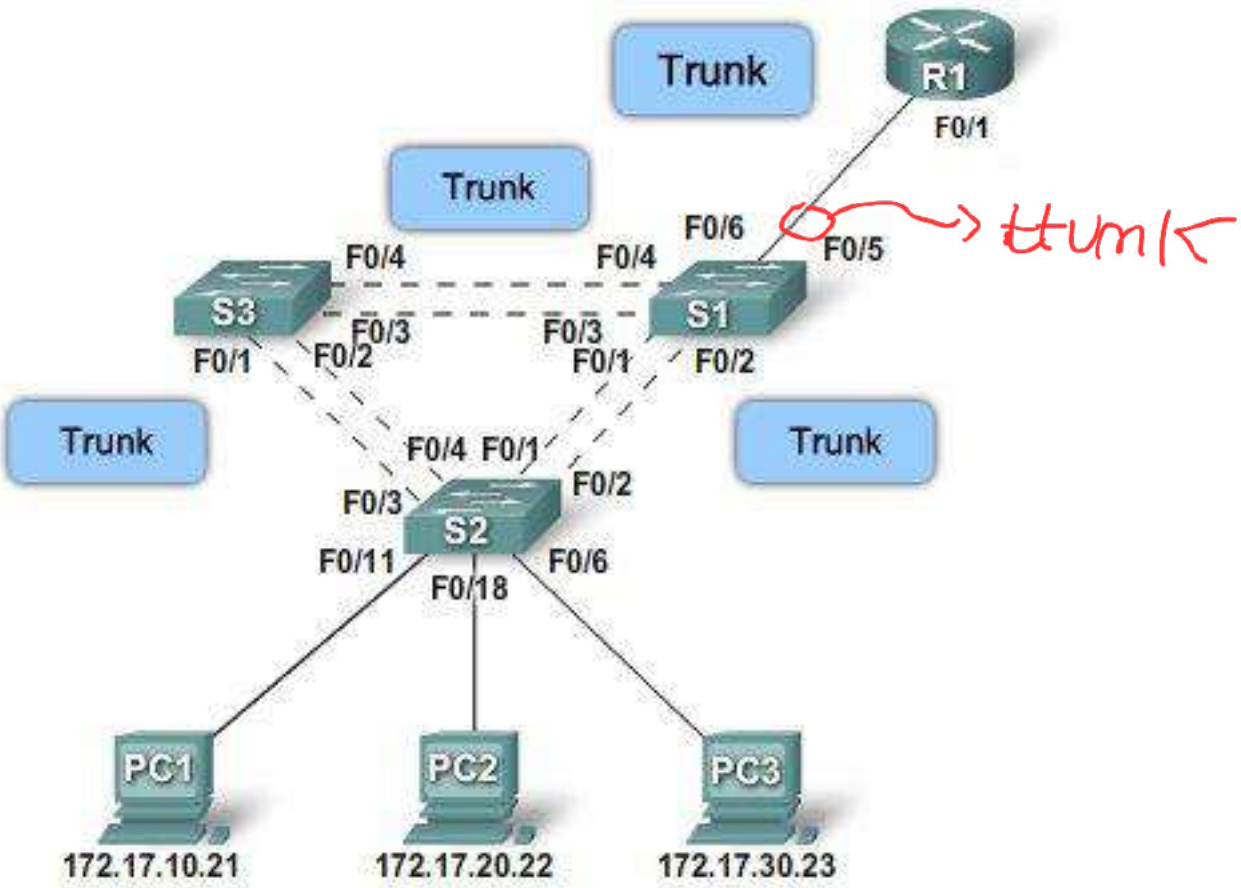
"Router-on-a-stick" Inter-VLAN

Ri —— Si
one!

- Even in this case, a router is present.
- The router is connected to a switch by means of a single physical interface.
- The physical interface of the router is split into virtual interfaces (**subinterfaces**): the number of virtual interfaces is equal to the number of VLANs able to communicate each other.
- Each virtual interface (subinterface) of the router is associated to a single VLAN (it must have an IP address of the VLAN block).
- The switch port connected to the router must be configured in **trunk** mode.

"Router-on-a-stick" Inter-VLAN: example (1/3)

- Only VLAN 10 and VLAN 30 are allowed to communicate.



R1 Subinterfaces

F0/0.10: 172.17.10.1

F0/0.30: 172.17.30.1

Switch S1 Ports

F0/1-F0/4 = Trunk

F0/5 = Trunk

Switch S2 Ports

F0/11 = VLAN10

F0/18 = VLAN20

F0/6 = VLAN30

F0/1-F0/4 = Trunk

"Router-on-a-stick" Inter-VLAN: example (2/3)

- **Router**: the interface connected to the switch must be split in two subinterfaces, one belonging to VLAN 10 and one to VLAN 30

```
R1(config)# interface Fa 0/0.10
R1(config-subif)# encapsulation dot1q 10
R1(config-subif)#ip address 172.17.10.1  255.255.255.0

R1(config)# interface Fa 0/0.30
R1(config-subif)# encapsulation dot1q 30
R1(config-subif)# ip address 172.17.30.1  255.255.255.0

R1(config)# interface Fa 0/0
R1(config-if)# no shutdown
```

two virtual interface

real interface

"Router-on-a-stick" Inter-VLAN: example (3/3)

- **Switch**: the port connected to the router is configured in trunk mode

```
S1(config)# interface Fa 0/5
S1(config-if)# switchport mode trunk
```

Antonio Cianfrani

Access Control List (ACL)

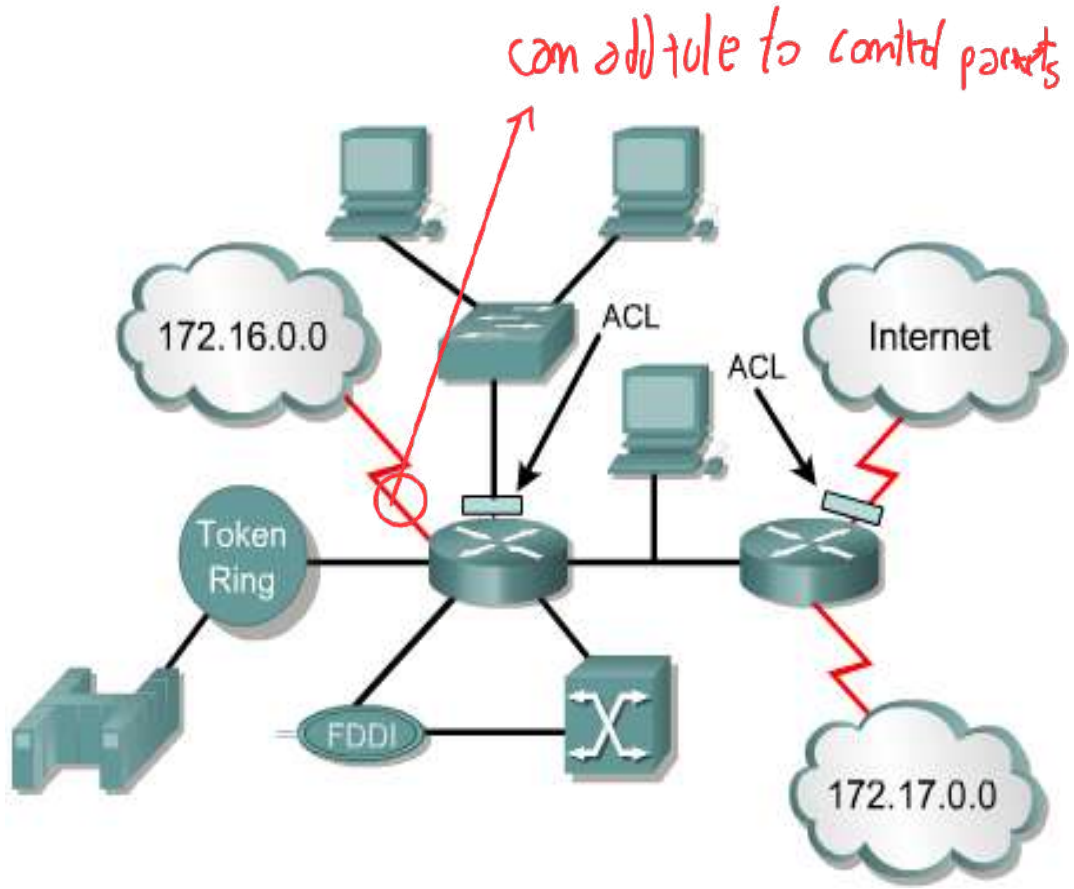
Index

- **ACL?**
- **How to configure**
  - Standard ACL
  - Extended ACL
  - Named ACL
- **Limiting the vty access**

ACL (1/3)

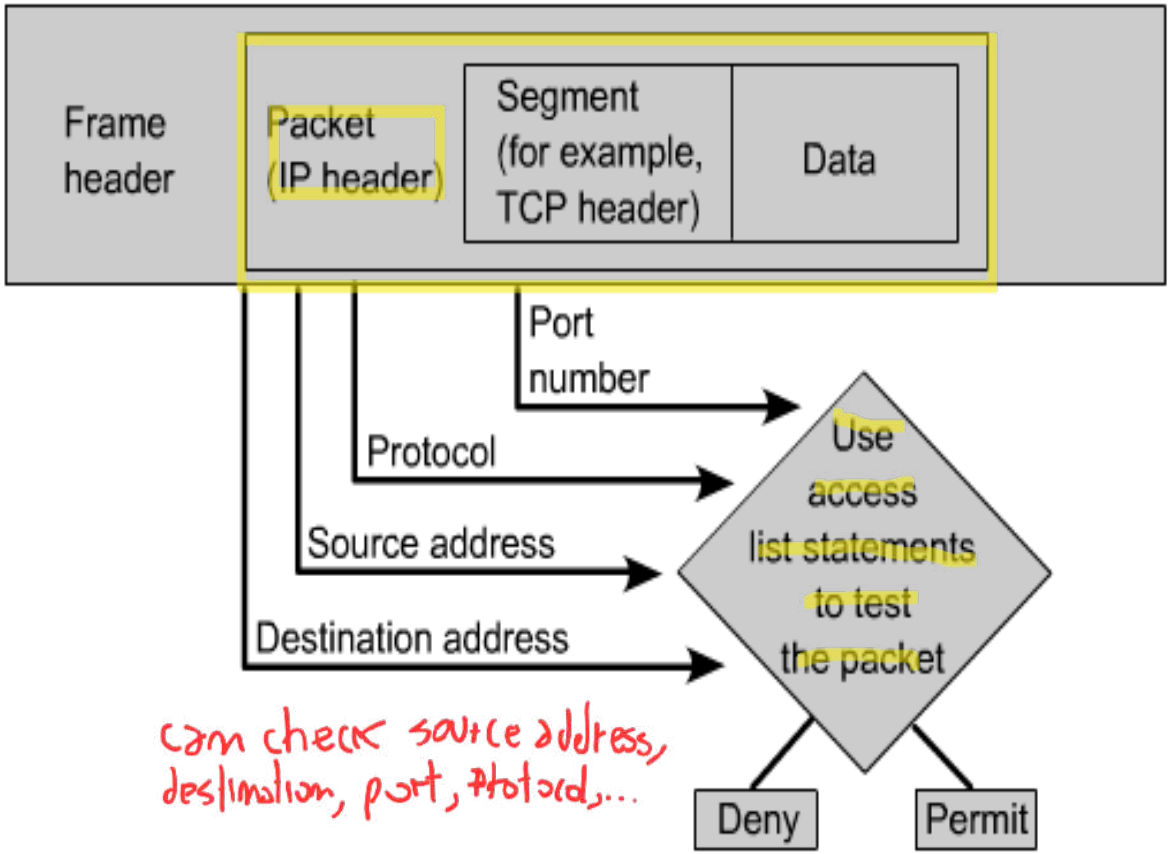
- Control lists applied to traffic incoming in / outgoing from a router
- Rules to determine if packets must be processed and forwarded or blocked and dropped by the router
- Configured on the router and applied to interfaces to control network access

also specify direction of traffic



ACL (2/3)

- Defined on the basis of the IP addresses, protocol, direction, port, etc…
- The ACL can be applied for incoming packets **(inbound)** or outgoing packets **(outbound)**



can check source address, destination, port, protocol,...

ACL (3/3)

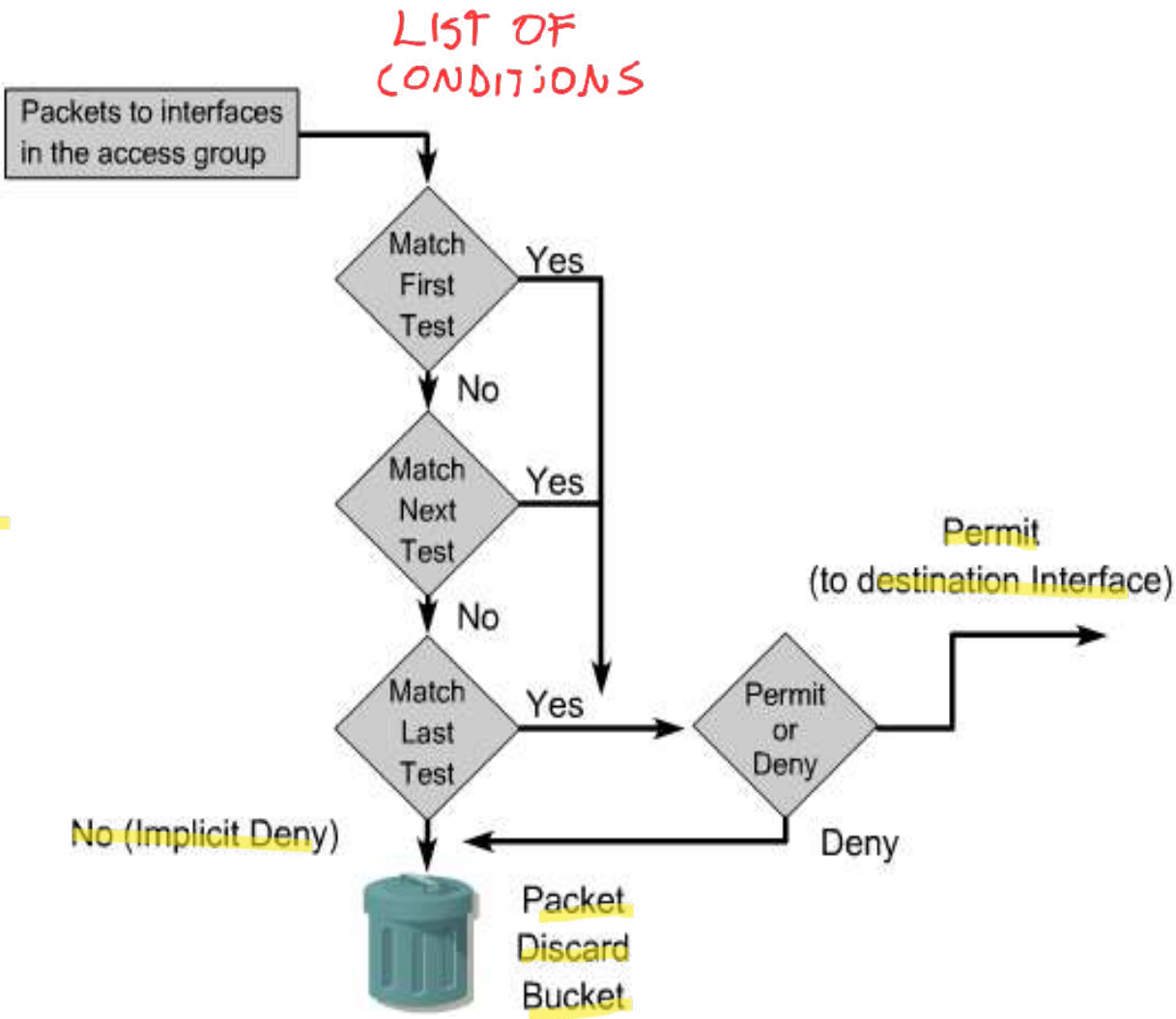
- Motivations
  - Limiting the network traffic
  - Controlling the traffic flows
  - Basic level of security
  - Allowing or denying the network access on the basis of traffic type
- If no ACLs are configured on a router, all the packets will be processed and forwarded

Inbound & Outbound interface

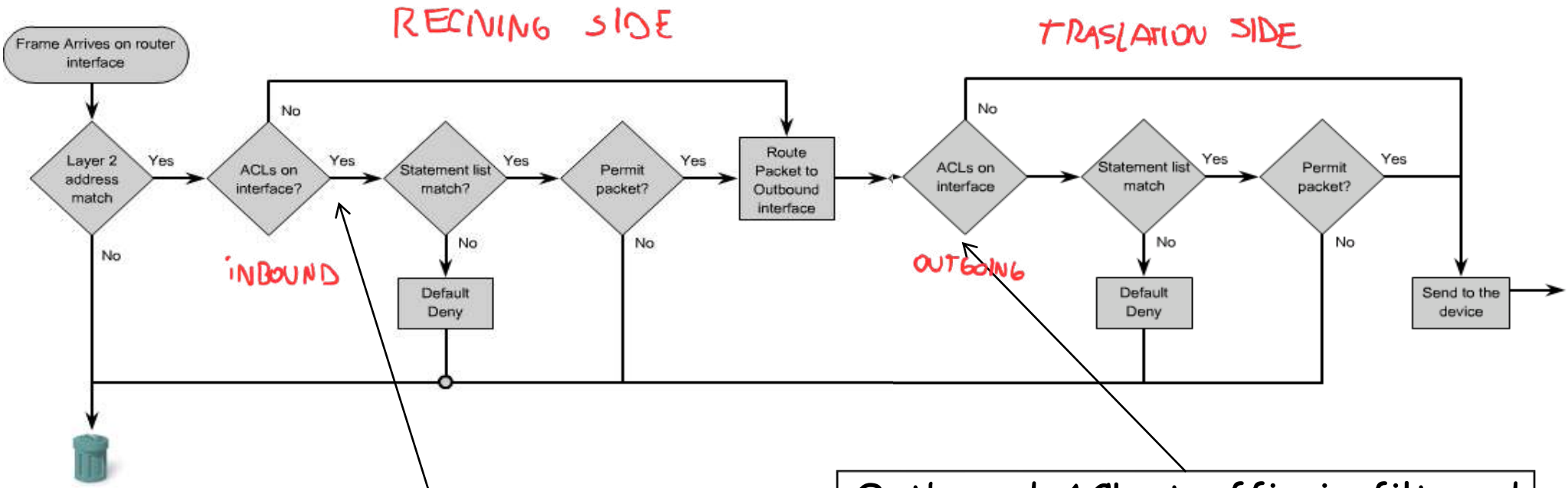
- The packets direction on an interface:
  - **Inbound**: incoming packets.
  - **Outbound**: outgoing packets.
- When associating an ACL to an interface, the direction must be specified.
- An ACL can be associated also to a sub-interface
  - If the sub-interfaces are defined on an interface, the ACL must be associated to subinterfaces, not to the original interface

ACL: list fo rules

- An ACL is an ordered list of rules about permitting or denying packets processing/forwarding on an interface
- The ordering of rules must be carefully defined!



ACL implementation



Inbound ACL: traffic is filtered before routing processing … and applied to all incoming packets

Outbound ACL: traffic is filtered after routing processing … and applied only to the packets forwarded through this interface

Creating an ACL (1/2)

- ACL are created in the global configuration mode.
- Three ACL types

  Source IP ADDRESS

  - **Standard**: packets filtering is based only on the source IP address.

    all other

  - **Extended:** packets filtering is based on source and destination IP address, protocol, and destination port.
  - **Named**: can be both Standard and Extended, are identified by a name and provide a more flexible configuration procedure. ⇝ defined by a name, more flexible configuration
- Classical Standard and Extended ACLs are identified by a unique (router-level) number (Numbered ACL)
  - Different ranges for standard and extended ACL

    1 – 99 101 – 199

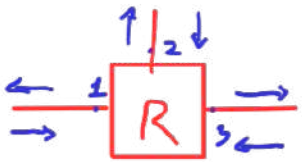
Creating an ACL (2/2)

access-list ... used for mat!

- The command to define a rule for an ACL identified by number X is "**access-list X**" followed by specific parameters
  - After the definition, an ACL must be associated to one (or more) interface(s) on a specific direction
- The command "**no access-list X**" removes the ACL x
- Le command to associate an ACL to an interface is "**ip access-group**", in the interface configuration mode:
  - **in** for inbound direction (applied to received packets)
  - **out** for outbound direction (applied to packet to be transmitted)

ACL example

6 different direction of traffic



can apply at max 6 acl
one for direction

```
Router(config)#no access-list 2
```

```
Router(config)#
access-list 2 deny   172.16.1.1
access-list 2 permit 172.16.1.0  0.0.0.255
access-list 2 deny   172.16.0.0  0.0.255.255
access-list 2 permit 172.0.0.0   0.255.255.255
interface ethernet 0
  ip access-group 2 in
```

in interface
must use IP access-group

The wildcard mask (1/2)

- The wildcard mask is a 32 bits string
- A wildcard define the matching of an ACL rule regarding IP addresses
  - 0 → the corresponding value of the IP address must be checked
  - 1 → the corresponding value of the IP address doesn't have to be checked

check only first two bytes

172.16.0.0 0.0.255.255:

172.16.2.3 → matching

172.15.0.1 → no matching

0 → check → not same

The wildcard mask (2/2)

- Two keywords to be used for ACL definition:
  - any
  - host

no check

- "**any**" means 255.255.255.255 for the wildcard mask:
  - Always a matching, independently of the IP address.
- "**host**" means 0.0.0.0 for the wildcard mask:
  - All the bits of the IP address must be checked.

check all bits

Verifying the ACL (1/3)

- "show ip interface"
- "show access-list"
- "show running-config"

Verifying the ACL (2/3)

```
Router#show ip interface
FastEthernet0/0 is up, line protocol is down
  Internet address is 192.168.1.1/24
  Broadcast address is 255.255.255.255
  MTU is 1500 bytes
  Helper address is not set
  Directed broadcast forwarding is disabled
  Outgoing access list is not set
  Inbound  access list is 2
Serial0/0 is down, line protocol is down
  Internet address is 200.200.2.1/24
  Broadcast address is 255.255.255.255
  MTU is 1500 bytes
  Helper address is not set
  Directed broadcast forwarding is disabled
  Outgoing access list is not set
  Inbound access list is 101
```

Verifying the ACL (3/3)

```
Router#show access-lists
Standard IP access list 2
 deny   172.16.1.1
 permit 172.16.1.0, wildcard bits 0.0.0.255
 deny   172.16.0.0, wildcard bits 0.0.255.255
 permit 172.0.0.0, wildcard bits 0.255.255.255
Extended IP access list 101
 permit tcp 192.168.6.0 0.0.0.255 any eq telnet
 permit tcp 192.168.6.0 0.0.0.255 any eq ftp
 permit tcp 192.168.0.0 0.0.0.255 any eq ftp-data
Router#
```

```
access-list 2 deny   172.16.1.1
access-list 2 permit 172.16.1.0 0.0.0.255
access-list 2 deny   172.16.0.0 0.0.255.255
access-list 2 permit 172.0.0.0 0.255.255.255
access-list 101 permit tcp 192.168.6.0 0.0.0.255 any
eq telnet
access-list 101 permit tcp 192.168.6.0 0.0.0.255 any
eq ftp
```

Standard ACL (1/3)

- A standard ACL filters on the basis of the source IP address
  - The access-list number must be in the range [1, 99] (also [1330, 1999] in last IOS version)
- The command syntax is :

```
Router(config)#access-list access-list-number
   {deny | permit} source [source-wildcard ]
```

- To remove the ACL:

```
Router(config)#no access-list access-list-number
```

Standard ACL (2/3)

```
access-list 2 deny   172.16.1.1
access-list 2 permit 172.16.1.0  0.0.0.255
access-list 2 deny   172.16.0.0  0.0.255.255
access-list 2 permit 172.0.0.0   0.255.255.255
```

- Access list number range of 1-99
- Filter only on source IP address
- Wildcard masks
- Applied to port closest to destination

Standard ACL (3/3)

Parameter	Description
access-list-number	Number of an ACL. This is a decimal number from 1 to 99 (for a standard IP ACL).
deny	Denies access if the conditions are matched.
permit	Permits access if the conditions are matched.
source	Number of the network or host from which the packet is being sent. There are two ways to specify the *source*: ·Use a 32-bit quantity in four-part, dotted- decimal format. ·Use the keyword **any** as an abbreviation for a *source* and *source-wildcard* of 0.0.0.0 255.255.255.55.
source-wildcard	(Optional) Wildcard bits to be applied to the source. There are two ways to specify the *source-wildcard*: ·Use a 32-bit quantity in four-part, dotted-decimal format. Place ones in the bit positions you want to ignore. ·Use the keyword **any** as an abbreviation for a *source* and *source-wildcard* of 0.0.0.0 255.255.255.255. (Optional) Causes an informational logging message about the packet that matches the entry to be sent to the console. (The level of messages logged to the console is controlled by the **logging console** command.)
log	The message includes the ACL number, whether the packet was permitted or denied, the source address, and the number of packets. The message is generated for the first packet that matches, and then at five-minute intervals, including the number of packets permitted or denied in the prior five-minute interval.

standard Exercise

better solution:

```
access-list 1 deny 192.168.30.128
access-list 1 permit any
interface Fa0/0
    ip access-group 1 in
```

50/10 out: correct but drop packet after, is not optimal choiche.

permit ip any any to permit all!

Standard ACL: summary

Step 1	Define the ACL by using the following command: `Router(config)#access-list access-list-number {permit | deny} {test-conditions}` A global statement identifies the ACL. Specifically, the 1-99 range is reserved for standard IP. This number refers to the type of ACL. In Cisco IOS Release 11.2 or newer, ACLs can also use an ACL name, such as education_group, rather than a number. The **permit** or **deny** term in the global ACL statement indicates how packets that meet the test conditions are handled by Cisco IOS software. **permit** usually means the packet will be allowed to use one or more interfaces that you will specify later.The final term or terms specifies the test conditions used by the ACL statement.
Step 2	Next, you need to apply ACLs to an interface by using the **access-group** command, as in this example: `Router(config-if)#{protocol} access-group access-list-number` All the ACL statements identified by *access-list-number* are associated with one or more interfaces. Any packets that pass the ACL test conditions can be permitted to use any interface in the access group of interfaces.

Extended ACL (1/4)

- The Extended ACLs provide a more accurate control on network traffic.
- The Extended ACLs allow to filter traffic considering **IP source and destination addresses**, **the protocol** and **the destination port number**.
  - Example: an extended ACL can permit e-mail traffic toward a specific destination and denies file transfers and web browsing toward the same destination.
- The command syntax is similar to the one of standard ACL: "*access-list*"
- The "*ip access-group*" command associate an extended ACL to an interface

Extended ACL (2/4)

- The ACL statements could be very long.
  - Can be simplified using the *host* and *any* options
- The port number field is optional:
  - equal (eq), not equal (neq), greater than (gt) and less than (lt).
- The extended ACL identifier must belong to the range [101,199].

Extended ACL (3/4)

```
Router(config)#access-list access-list-number {permit | deny}
protocol source
[source-mask destination destination-mask operator operand]
[established]
```

Paramter	Description
access-list-number	Identifies the list using a number in the range 100 to 199.
permit | deny	Indicates whether this entry allows or blocks the specified address.
protocol	The protocol, such as IP, TCP, UDP, ICMP, GRE, or IGRP. PING
source and **destination**	Identifies source and destination addresses.
source-mask and *destination-mask*	Wildcard mask; zeros indicate positions that must match, ones indicate do not care positions.
operator operand	lt, gt, eq, neq (less than, greater than, equal, not equal), and a port number.
established	Allows TCP traffic to pass if the packet uses an established connection (for example, has ACK bits set).

DNS → UDP
ICMP → IP

Extended ACL (4/4)

all protocol of TCP

Extended ACL

source IP

wildcard source

equal

```
access-list 114 permit tcp 172.16.6.0 0.0.0.255 any eq telnet
access-list 114 permit tcp 172.16.6.0 0.0.0.255 any eq ftp
access-list 114 permit tcp 172.16.6.0 0.0.0.255 any eq ftp-data
```

destination

- Access list number range of 100-199
- Source destination IP address
- Layer 4 protocol number
- Applied to port closest to source host

Port Numbers

```
R1(config)#access-list 101 permit tcp any eq ?

  <0-65535> Port number
  bgp Border Gateway Protocol (179)
  chargen Character generator (19)
  cmd Remote commands (rcmd, 514)
  daytime Daytime (13)
  discard Discard (9)
  domain Domain Name Service (53)
  drip Dynamic Routing Information Protocol (3949)
  echo Echo (7)
  exec Exec (rsh, 512)
  finger Finger (79)
  ftp File Transfer Protocol (21)
  ftp-data FTP data connections (20)
  gopher Gopher (70)
  hostname NIC hostname server (101)
  ident Ident Protocol (113)
  irc Internet Relay Chat (194)
  klogin Kerberos login (543)
  kshell Kerberos shell (544)
  login Login (rlogin, 513)
  lpd Printer service (515)
  nntp Network News
  Transport Protocol (119)
  pim-auto-rp PIM Auto-RP (496)
  pop2 Post Office Protocol v2 (109)
  pop3 Post Office Protocol v3 (110)
  smtp Simple Mail Transport Protocol (25)
  sunrpc Sun Remote Procedure Call (111)
  syslog Syslog (514)
  tacacs TAC Access Control System (49)
  talk Talk (517)
  telnet Telnet (23)
  time Time (37)
  uucp Unix-to-Unix Copy Program (540)
  whois Nicname (43)
  www World Wide Web (HTTP, 80)
```

dns

Named ACL (1/3)

- The named ACLs are identified by a name.
- A named ACL can be standard or extended
- named ACLs can be modified with no need of starting from scratch
  - It is possible to remove a single statement and to add a new one in any position
- The command to create a named ACL is "ip access-list"
  - An ACL sub-configuration CLI exists

Named ACL (2/3)

```
ip access-list {extended|standard} name
```

new sub-configuration

```
router(config-ext-nacl)#permit|deny protocol source
source-wildcard [operator [port]] destination
destination-wildcard [operator [port]] [established]
[precedence precedence] [tos tos] [log] [time-range time-
range-name]
```

```
ip interface ethernet0/5
ip address 192.168.5.1 255.255.255.0
ip access-group Internetfilter out
ip access-group marketinggroup in
...
ip access-list standard Internetfilter
permit 10.1.1.1
deny any
ip access-list extended marketing_group
permit tcp any 172.30.0.0.0.255.255.255 eq telnet
deny udp any any
deny udp any 171.30.0.0.0.255.255.255 1t 1024
deny ip any log
```

Named ACL (3/3)

```
Rt1(config)#ip access-list extended server-access
Rt1(config-ext-nacl)#permit TCP any host 131.108.101.99 eq
smtp
Rt1(config-ext-nacl)#permit UDP any host 131.108.101.99 eq
domain
Rt1(config-ext-nacl)#deny ip any any log
Rt1(config-ext-nacl)#^Z
Applying the named list:
Rt1(config)#interface fastethernet 0/0
Rt1(config-if)#ip access-group server-access out
Rt1(config-if)#^Z
```

have one number
10, 20, 30 each

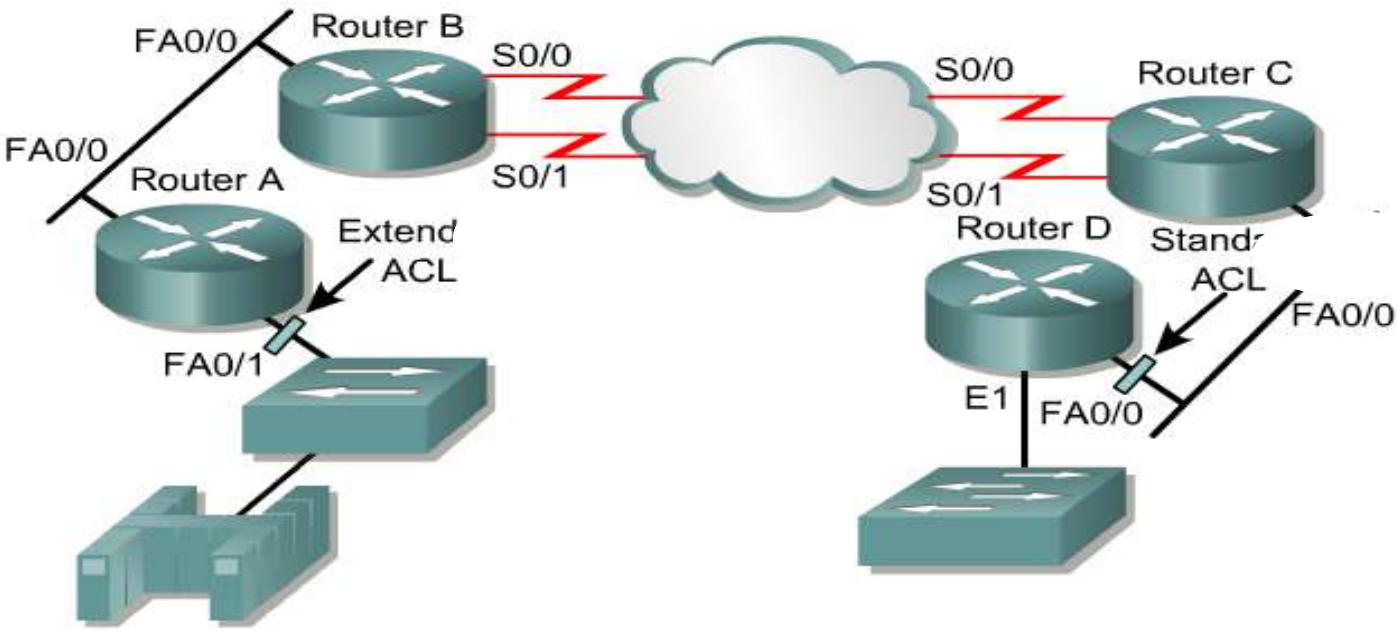
Modificare una Named ACL

```
R1(config)#ip access-list extended SERVER-ACCESS
R1(config-ext-nacl)# no 20
R1(config-ext-nacl)# 20 permit ip host 192.168.1.77 any
R1(config-ext-nacl)# end
R1#show access-lists
Extended IP access list SERVER-ACCESS
    10 permit ip host 192.168.1.66 host 192.168.3.75
    20 permit ip host 192.168.1.77 any
    30 deny ip 192.168.1.0 0.0.0.255 host 192.168.3.75
```

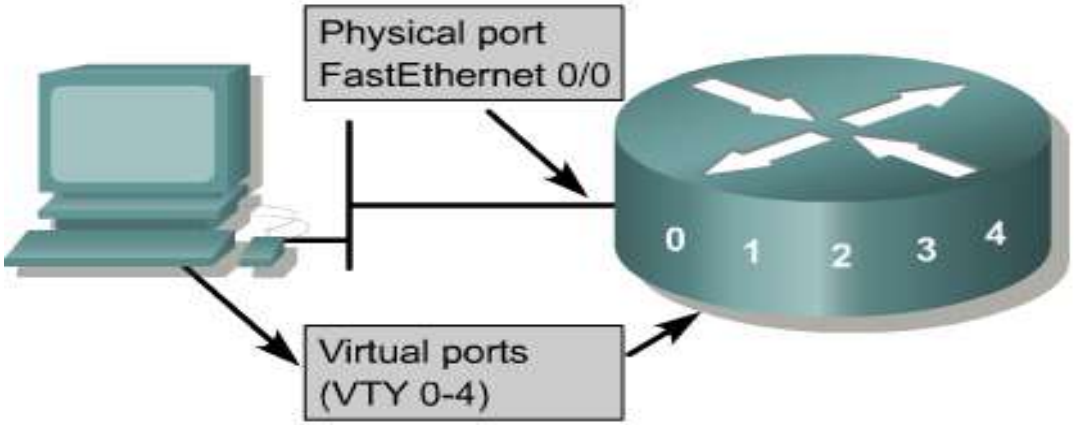
```
R1(config)#ip access-list extended SERVER-ACCESS
R1(config-ext-nacl)# 25 deny ip host 192.168.1.88 any
R1(config-ext-nacl)# end
R1#show access-lists
Extended IP access list SERVER-ACCESS
    10 permit ip host 192.168.1.66 host 192.168.3.75
    20 permit ip host 192.168.1.77 any
    25 deny ip host 192.168.1.88 any
    30 deny ip 192.168.1.0 0.0.0.255 host 192.168.3.75
```

Locating the ACL

- General rules:
  - The standard ACLs should be put as close as possible to the destination (since it is not possible to specify the destination)
  - The extended ACLs should ne put as clos as possible to the source (to filter earlier the traffic, and so reducing routers work)



Limiting the telnet access



Cisco - Hyperterminal

```
Creating the standard list:

Rt1(config)#access-list 2 permit 172.16.1.0 0.0.0.255
Rt1(config)#access-list 2 permit 172.16.2.0 0.0.0.255
Rt1(config)#access-list 2 deny any

Applying the access list:

Rt1(config)#line vty 0 4
Rt1(config)#login
Rt1(config)#password secret
Rt1(config)#access-class 2 in
```

use this STANDARD list to limit traffic to telnet

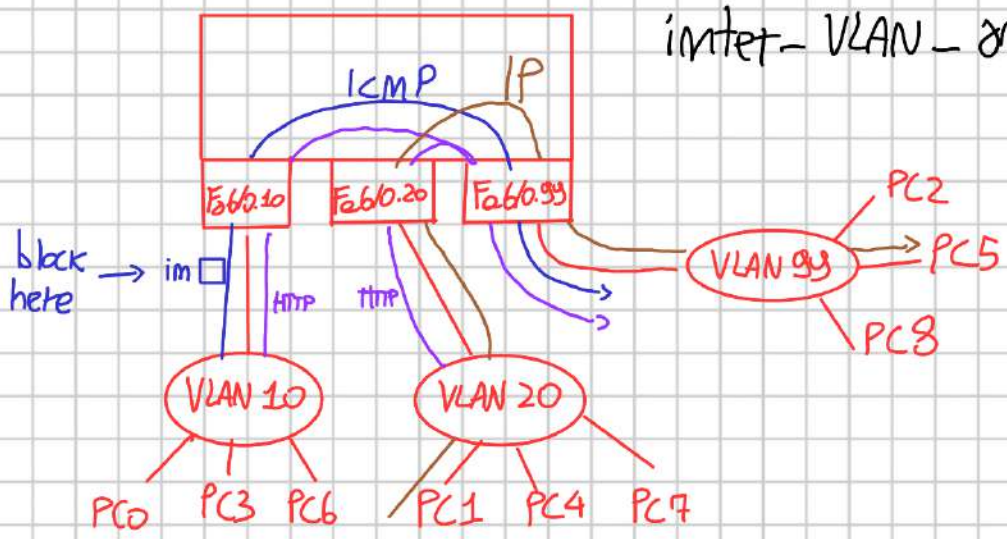
only this traffic can access telnet

OPIS

OPIS CODE:

SAXH4DZL

inter-VLAN and ACL



must block ICMP and HTTP in .10 in to VLAN 99

must block HTTP and IP to PC5-(PC4) in .20 in to VLAN 99

in 1. could use:

deny ICMP any 192.168.99.0 0.0.0.255

↳ all traffic

For telnet:

access-list 1 permit 11.10.0.0 0.0.255.255

access-list 1 permit 192.168.99 0.0.0.255

line vty 0 4

(...)# access-class 1 in

exit

Antonio Cianfrani

LAN Security

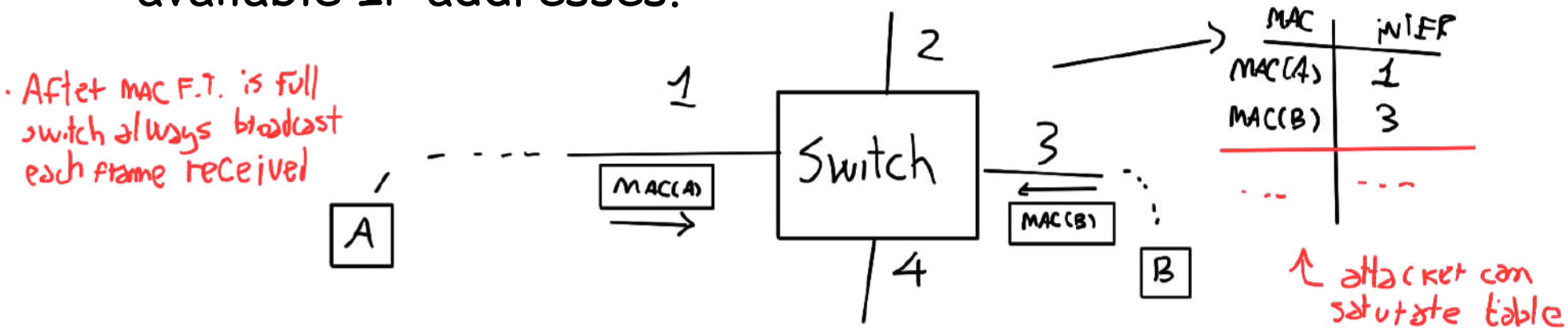
Security attacks in LAN (1/2)

- The classical attack in a LAN is the **MAC Address Flooding**.
- It exploits the security weakness of MAC forwarding table learning mechanism:
  - If an incoming frame with a new MAC source address is received, the switch add a row in the forwarding table
  - If an incoming frame has a destination MAC address not present in the forwarding table, the switch acts as an hub
  - The forwarding tables have a limited size
- MAC Address Flooding:
  - Frames with artificial source MAC address → the forwarding table is saturated → frames with new MAC destination address are forwarded in broadcast

Security attacks in LAN (2/2)

- **DHCP Spoofing:** a malicious DHCP server is inserted in the LAN, so that fake info (default gateway) are notified to LAN hosts. This is a man-in-the-middle attack

- **DHCP starvation:** attack to the DHCP servers, sending a huge amount of DHCP requests so that to use all the available IP addresses.



Port Security (1/4)

- **Port Security**: option to be configured on switch interface/s to increase the security level of the network
- The idea of Port Security is to limit the end devices that can be connected to a specific switch interface
- The security policy is based on the source MAC address of incoming packets and on the number of different source MAC addresses allowed on the interface.

can limit messages that can send a certain MAC

Port Security (2/4)

- If a frame having a MAC source address not allowed is received, the interface switch to ***Violation Mode:***
  - ✓ Shutdown by default (*error disabled state*).
  - ✓ To recover from the *error disabled state* it is necessary to manually shutdown the interface
- It is possible to allow the access to a single MAC address or to a range of MAC addresses.
- The association among the interface and the allowed MAC address/es can be dynamic or static
- First step of the configuration: enable port-security

  Switch(config-if)# switchport port-security

Port Security (3/4)

- Three different Port Security configuration modes:
  - Static: the allowed MAC address/es are statically configured by the LAN administrator with the command

Switch(config-if)# switchport port-security mac-address *mac-address*

- To look at the MAC address of a PC, from the Command Prompt: **ipconfig /all**

Command Prompt

```
C:\>ipconfig \all
Invalid Command.

C:\>ipconfig /all

FastEthernet0 Connection:(default port)

   Connection-specific DNS Suffix..:
   Physical Address................: 0060.2F52.1EAA
   Link-local IPv6 Address.........: FE80::260:2FFF:FE52:1EAA
   IP Address......................: 0.0.0.0
   Subnet Mask.....................: 0.0.0.0
   Default Gateway.................: 0.0.0.0
   DNS Servers.....................: 0.0.0.0
   DHCP Servers....................: 0.0.0.0
   DHCPv6 Client DUID..............: 00-01-00-01-C9-28-3C-D9-00-60-2F-52-1E-AA
```

must do port-security on all access-port of a VLAN, if asked in exam

Port Security (4/4)

- Dynamic: the allowed MAC addresses are learned dynamically up to a fixed number (1 by default) and saved only in the secure MAC address table

only how many allowed

- Sticky Dynamic: the allowed MAC addresses are learned dynamically up to a fixed number and saved in the secure MAC address table and in the running configuration file.

```
S1#configure terminal
S1(config)#interface fastEthernet 0/18
S1(config-if)#switchport mode access
S1(config-if)#switchport port-security
S1(config-if)#switchport port-security maximum 50
S1(config-if)#switchport port-security mac-address sticky
S1(config-if)#end
```

if not default is 1

sticky

Port Security: violation

- Once a violation occurs on a port, the port will transition to an error disabled state.
- To recover from an error disabled state, a ***shutdown*** command and then a ***no shutdown*** on the interface (manual intervention by an administrator) must be executed.
- Error disabled ports can be configured to automatically recover from port security violations:

S1(config-if)# errdisable recovery interval *Time_Interval*

- The *Time_Interval* is an integer value in the range [30-86400] seconds.

Port Security checking (1/2)

```
switch#show port-security interface fastEthernet 0/18
Port Security              : Enabled
Port Status                : Secure-down
Violation Mode             : Shutdown
Aging Time                 : 0 mins
Aging Type                 : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses      : 1
Total MAC Addresses        : 1
Configured MAC Addresses   : 0
Sticky MAC Addresses       : 0
Last Source Address:Vlan   : 0000.0000.0000:0
Security Violation Count   : 0
```

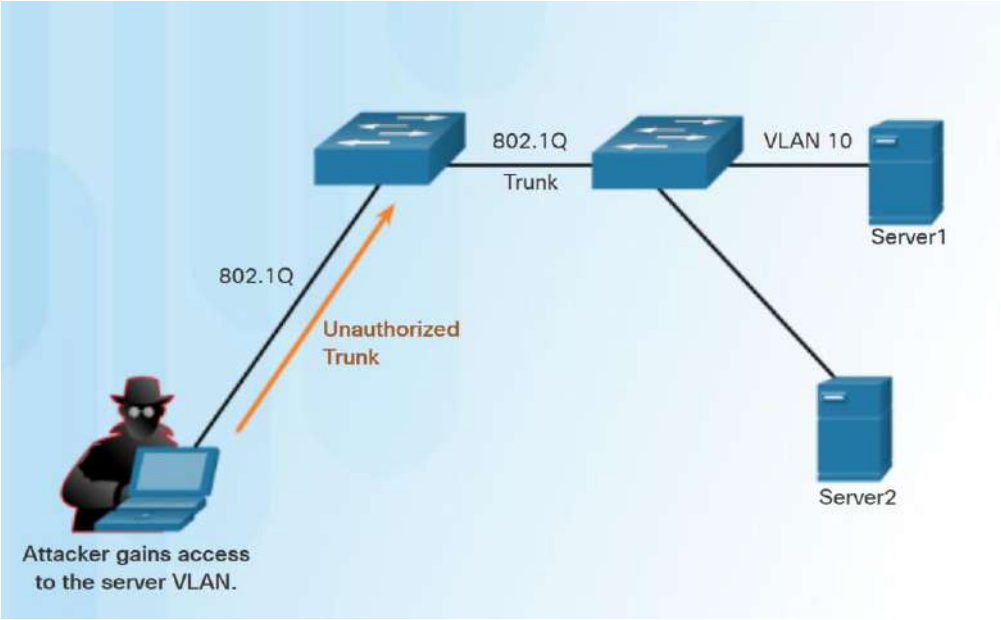
Port Security checking (1/2)

to see table

```
switch#show port-security address
        Secure Mac Address Table
-----------------------------------------------------------------------
Vlan   Mac Address       Type                  Ports     Remaining Age (mins)
99     0050.BAA6.06CE    SecureConfigured      Fa0/18    -
-----------------------------------------------------------------------
Total Addresses in System (excluding one mac per port)     : 0
Max Addresses limit in System (excluding one mac per port) : 8320
```

VLAN Attacks

- Attacker can gain VLAN access by configuring a host to spoof a switch:
  - Creating a trunk with an "unsecured" switch
  - Exploiting the "auto trunking" (DTP) configured by default on switch ports



VLAN Attacks

- Methods to mitigate VLAN attacks:
  - Explicitly configure access ports.
  - Disable auto negotiate on trunks.
  - Manually enable trunk links.
  - Disable unused ports, make them access ports, and assign to a black hole VLAN.
  - Change the default native VLAN.
  - Implement port security.

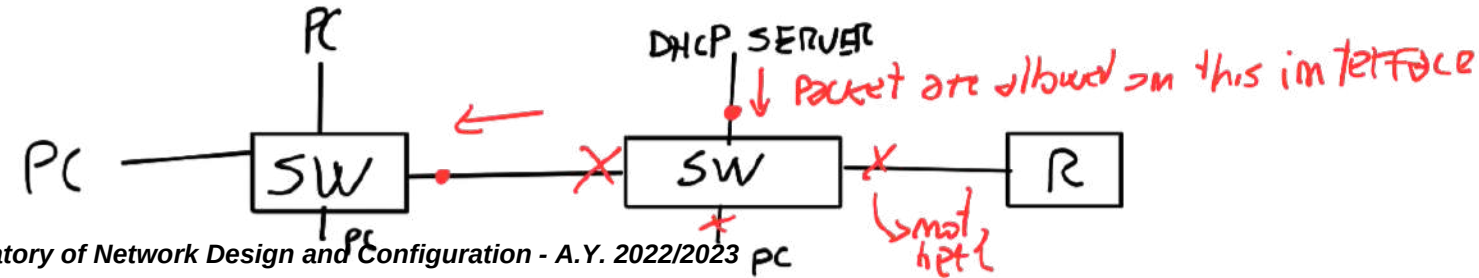
↳ exam: VLAN or interface asked

DHCP Attacks

- DHCP spoofing attack - An attacker configures a fake DHCP server on the network to issue IP addresses to clients.
- DHCP starvation attack - An attacker floods the DHCP server with bogus DHCP requests and leases all of the available IP addresses. This results in a denial-of-service (DoS) attack as new clients cannot obtain an IP address.
- Methods to mitigate DHCP attacks:
  - Configure **DHCP snooping**
  - Configure **port security**

DHCP Snooping

- **With DHCP snooping enabled on an interface, the switch will deny packets containing:**
  - Unauthorized DHCP server messages coming from an untrusted port.
- **DHCP snooping recognizes two types of ports:**
  - **Trusted DHCP ports** - Only ports connecting to upstream DHCP servers should be trusted. connect to DHCP of upstream
  - **Untrusted ports** - These ports connect to hosts that should not be providing DHCP server messages.



DHCP Snooping configuration

- Step 1. Enable DHCP snooping globally

```
S1(config)# ip dhcp snooping
```

- Step 2. Enable DHCP snooping on selected VLANs (at least one!)

if not have use 1

```
S1(config)# ip dhcp snooping vlan vlan-id
```

- Step 3. Configure trusted interfaces (untrusted is default)

```
S1(config-if)# ip dhcp snooping trust
```

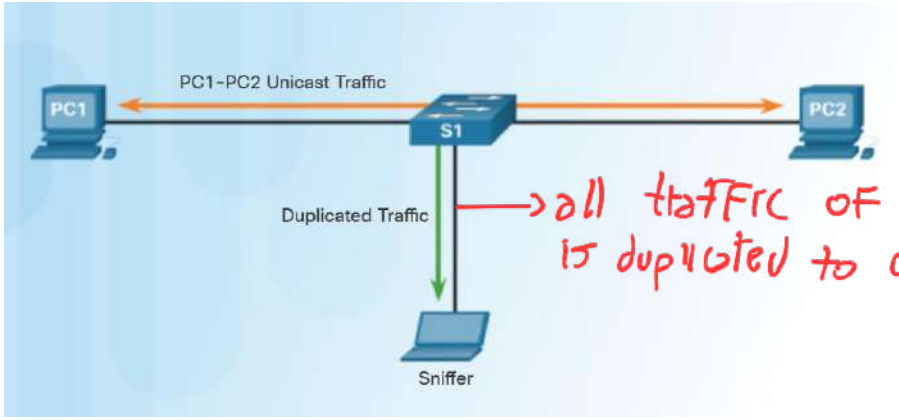
- Step 4. Disable option 82 insertion by switch

```
S1(config)# no ip dhcp snooping information option
```

NOT IN EXERCISES

Port Mirroring

- Port mirroring allows a switch to copy and send Ethernet frames from specific ports to a selected destination port (usually connected to a packet analyzer or to an Intrusion Detection System).



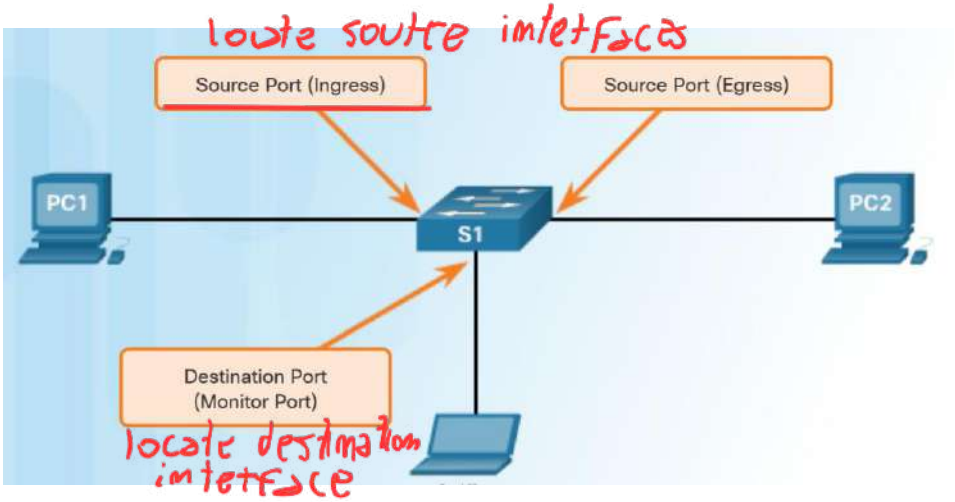
→ all traffic of specific interfaces is duplicated to other interfaces

- SPAN is a type of port mirroring
- SPAN can be implemented in two different ways: Local SPAN or Remote SPAN.

Local SPAN

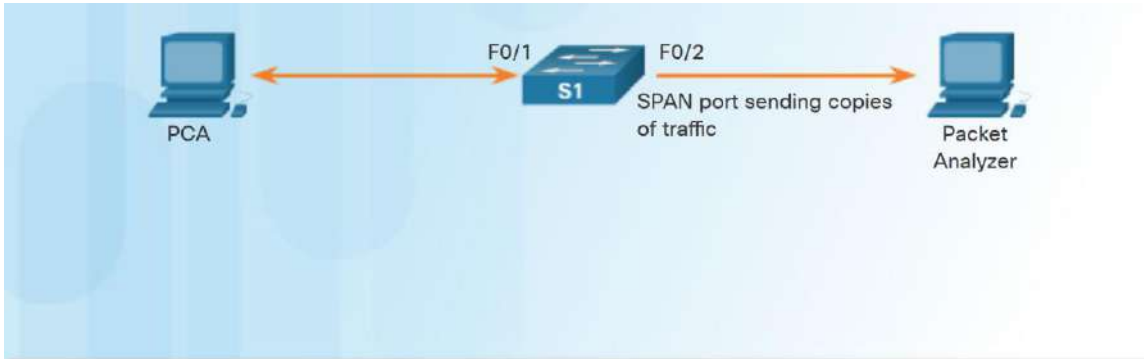
- Local SPAN: mirroring is performed on the same switch.
- A SPAN session is the association between source ports (or VLANs) and a destination port.
- The destination port is no longer a normal switch port. Only monitored traffic passes through that port.

Term	Definition
Ingress traffic	This is traffic that enters the switch.
Egress traffic	This is traffic that leaves the switch.
Source (SPAN) port	This is a port that is monitored with use of the SPAN feature.
Destination (SPAN) port	This is a port that monitors source ports, usually where a packet analyzer, IDS or IPS is connected. This port is also called the monitor port.
SPAN session	This is an association of a destination port with one or more source ports.
Source VLAN	This is the VLAN monitored for traffic analysis.



Configuring local SPAN

- A session number is used to identify a local SPAN session.
- The command ***monitor session*** configure SPAN with a source port and a destination port.
- A VLAN can be specified instead of a physical port.
- The ***show monitor*** command is used to check the SPAN sessions.



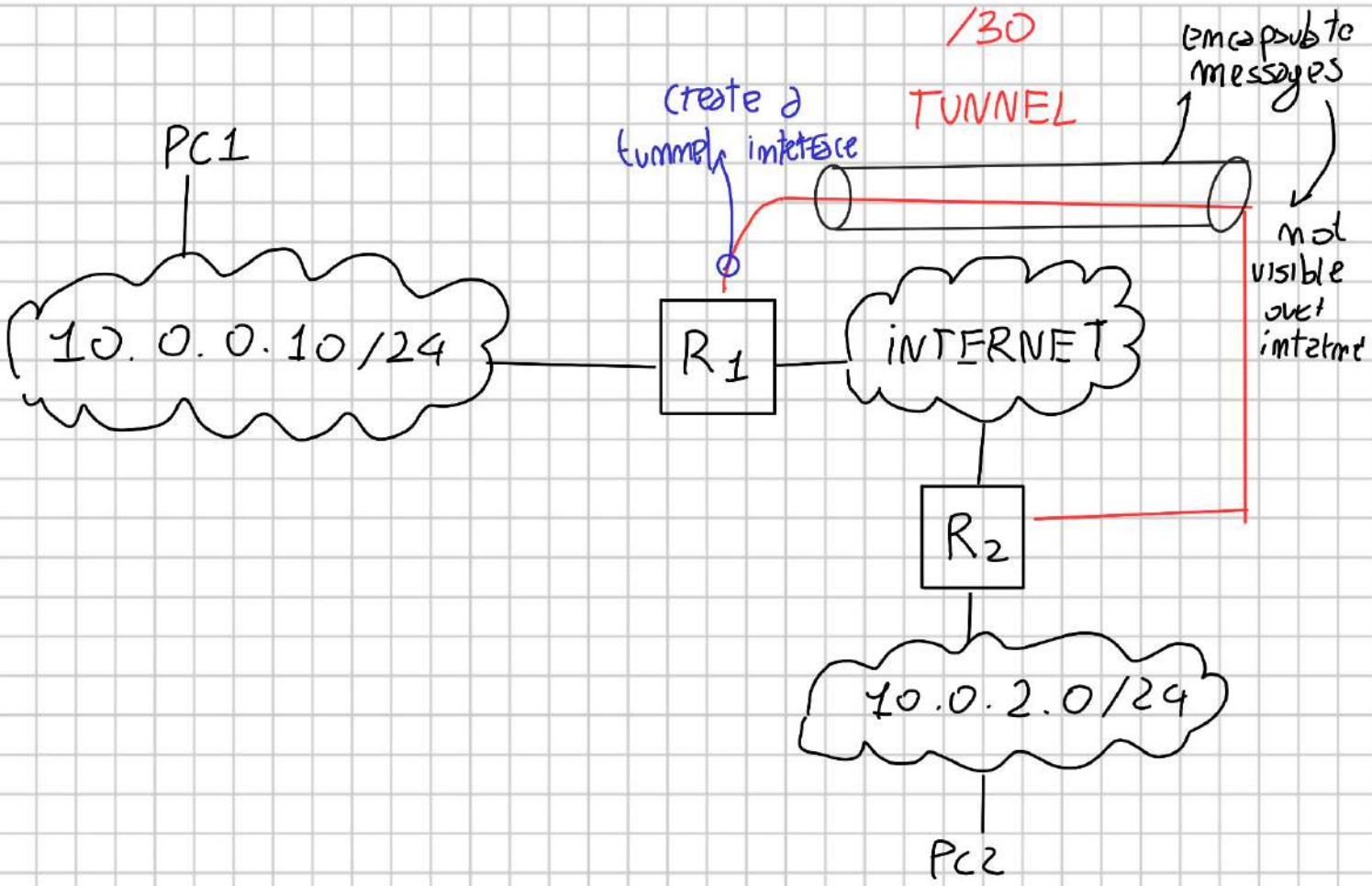

```
S1(config)# monitor session 1 source interface fastethernet 0/1
S1(config)# monitor session 1 destination interface fastethernet 0/2
```

```
S1# show monitor
Session 1
---------
Type                    : Local Session
Source Ports            :
    Both                : Fa0/1
Destination Ports       : Fa0/2
    Encapsulation       : Native
          Ingress       : Disabled


S1#
```

Virtual Private Network (VPN)

- A VPN is a private network created over a public network (Internet) exploiting the tunneling mechanism.
- Tunneling: encapsulation of a packet (frame) with an outer header. Many different tunneling mechanisms have been defined
- A VPN can also provide security functions, such as encryption: IPsec.
- A VPN gateway is necessary: it could be a router or a dedicated device.
- The VPN gateway is responsible for encapsulating and encrypting outbound traffic (and decapsulating and decrypting inbound traffic).

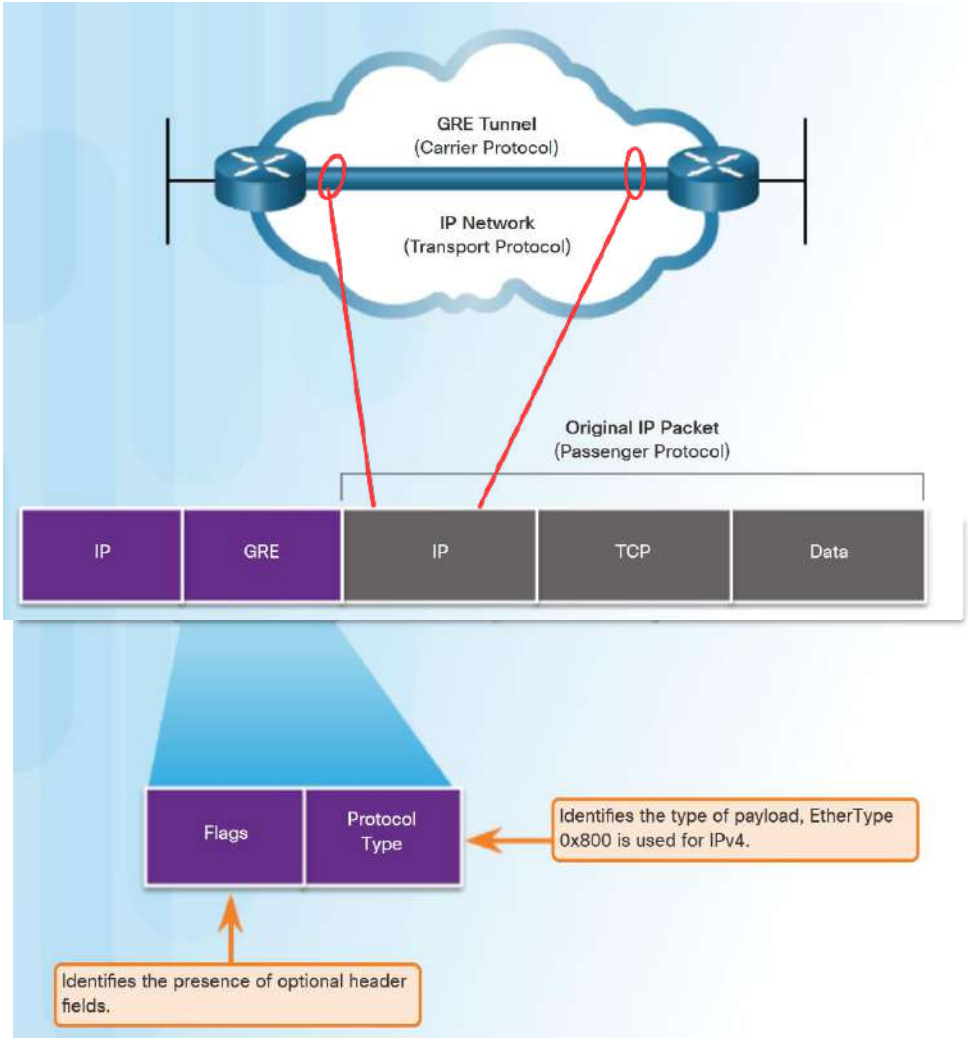


- Want PC1 to comunicate with PC2 without using NAT because are two private LAN
- Must configure R1 and R2 for implementing tunneling with encapsulation, adding extra header having a src and dst addresses that of router R1 and R2.

tunnel interface must have an ip-address can use for example a private ip-address with /30 and destination must use public ip-address of other router

Generic Routing Encapsulation (GRE)

- Generic Routing Encapsulation (GRE) is a non-secure, site-to-site VPN tunneling protocol.
- GRE is defined as an IETF standard (RFC 2784).
- In the outer IP header, 47 is used in the protocol field.
- GRE encapsulation uses a protocol type field in the GRE header to support the encapsulation of any OSI Layer 3 protocol.



Configuring GRE (1/2)

- Create a tunnel interface (not all router's models support it!).

  R(config)# interface Tunnel X

- Configure GRE as the tunnel interface mode.

  R(config-if)# tunnel mode gre ip

- Assign an IP address to the tunnel interface (private address)
- Specify the tunnel source physical interface.

  R(config-if)# tunnel source *InterfaceID*

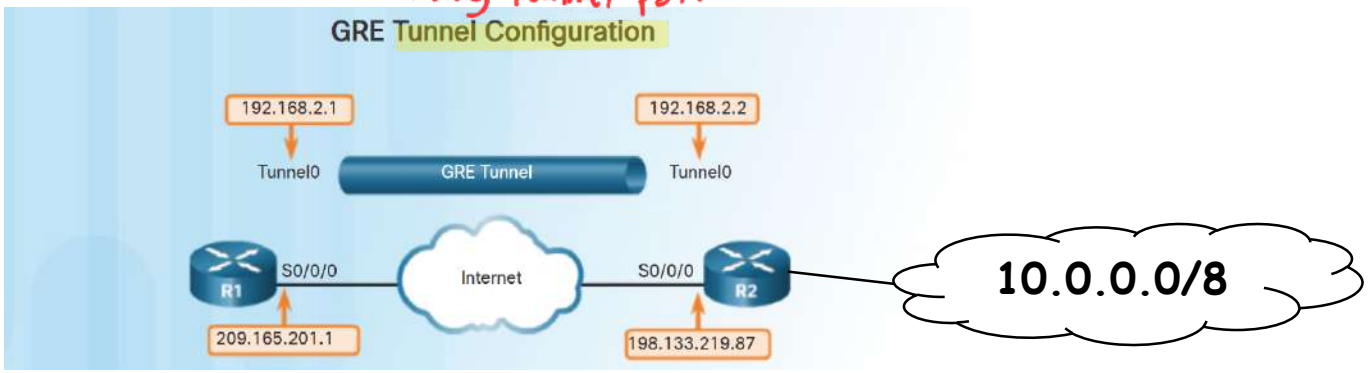
- Specify the tunnel destination IP address (IP address of the physical interface of the tunnel end point).

  R(config-if)# tunnel destination *IP_Address*

- Routing IP via tunnel interface!

Configuring GRE (2/2)

only tunnel part




```
R1(config)# interface Tunnel 0
R1(config-if)# tunnel mode gre ip
R1(config-if)# ip address 192.168.2.1 255.255.255.252
R(config-if)# tunnel source S0/0/0
R(config-if)# tunnel destination 198.133.219.87
R1(config)# ip route 10.0.0.0 255.0.0.0 192.168.2.2
```

public

static route to other LAN

other tunnel 0

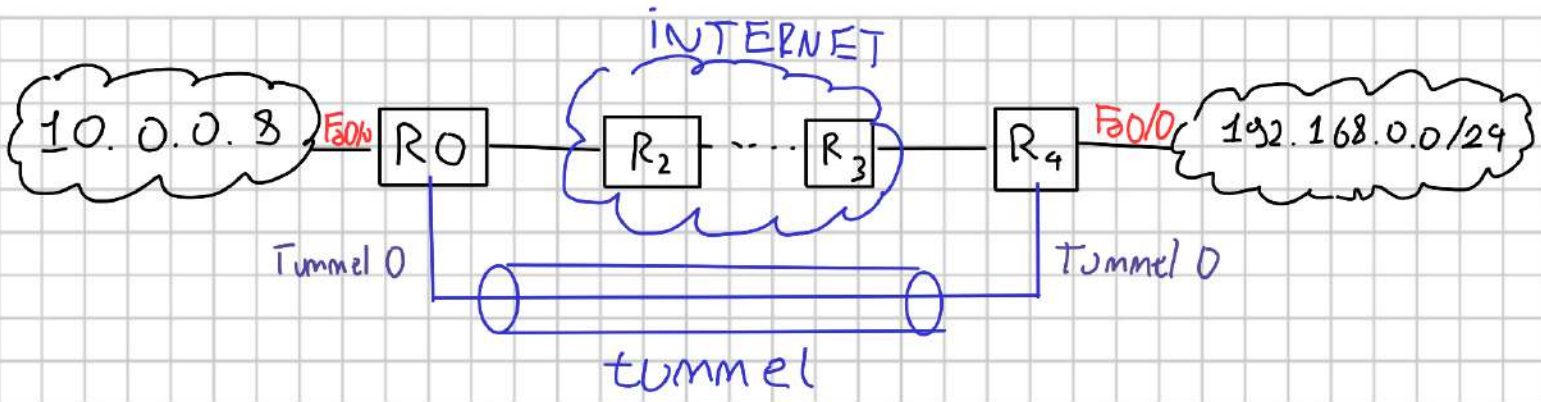
Verifying GRE

```
R1# show ip interface brief | include Tunnel

Tunnel0              192.168.2.1     YES manual up      up
```

```
R1# show interface Tunnel 0
Tunnel0 is up, line protocol is up
  Hardware is Tunnel
  Internet address is 192.168.2.1/24
  MTU 17916 bytes, BW 100 Kbit/sec, DLY 50000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation TUNNEL, loopback not set
  Keepalive not set
  Tunnel source 209.165.201.1, destination 209.165.201.2
  Tunnel protocol/transport GRE/IP

<output omitted>
```



- No Nat - configuted, and no OSPF on Fa 0/0, two privat LAN can not comunicate
- Can use Fot IP of tunnel 192.168.10.0/30

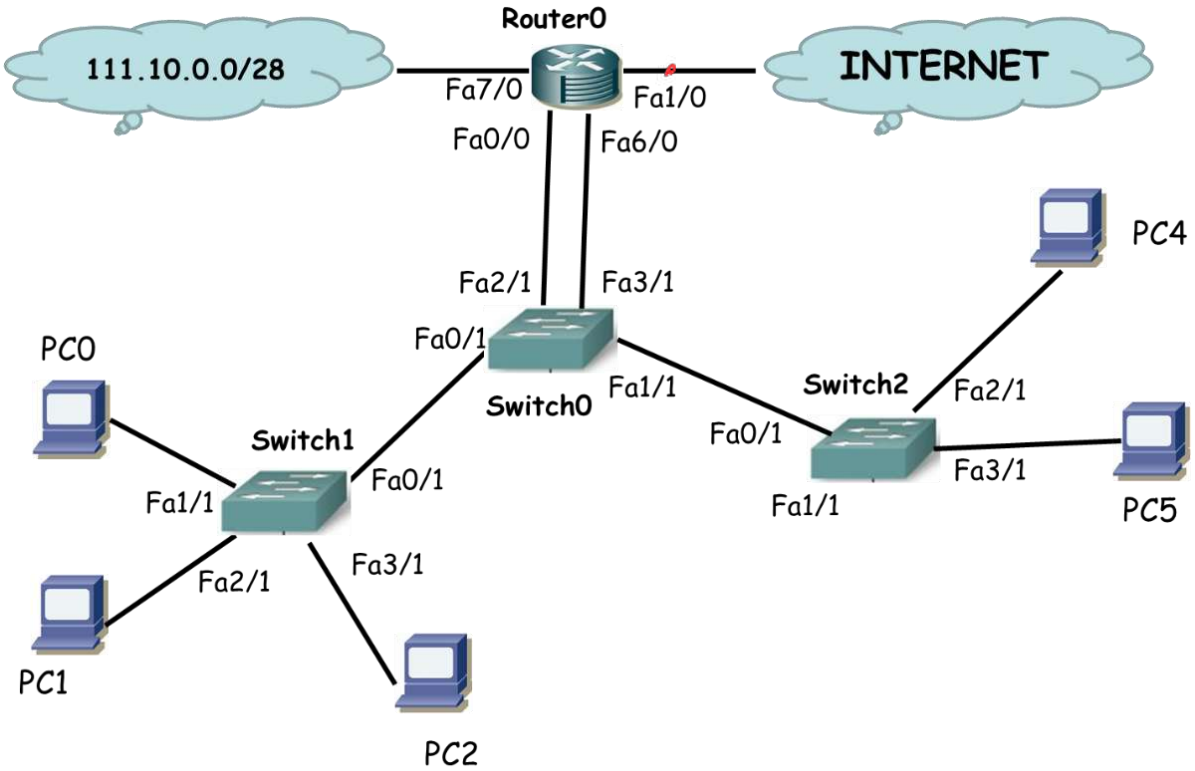
Exercitation II part - Laboratory of Network Design and Configuration

The student has to configure:

1. IP addresses of PC considering the following table:

Host	# VLAN	VLAN address block
PC0, PC4	10	11.0.0.0/16
PC1, PC5	30	30.0.0.0/18
PC2	99	99.0.0.0/26

2. VLANs on switches. IP OF SWITCHES
3. VLAN 99 as Management VLAN. All Switches must be accessible from PCs of management VLAN.
4. VLAN 10 as Native VLAN for all network trunks.
5. Traditional inter-VLAN routing among hosts of VLAN 10 and VLAN 30.
6. Sticky dynamic port security on interfaces of switch Switch2, fixing the maximum number of allowed MAC addresses to 2.
   only on access port Fa2/1, Fa3/1
   plug usi
7. ACLs so that only these traffic flows are blocked:
   a. HTTP traffic from Internet to VLAN 10 hosts;
   b. ICMP traffic from hosts of network 111.10.0.0/28 to PC0;
   c. UDP traffic from PC5 to PC4.



1. Configure vlan on all switchs

accessible and configurable ⟶ telnet + enable

accessible ⟶ telnet access