

Adrian PECULEA Bogdan IANCU Sorin BUZURA Vlad RAȚIU

Coordinators: Vasile Teodor DĂDĂRLAT, Emil CEBUC

Computer Networks

Practical activities



**UTPRESS
Cluj-Napoca, 2023
ISBN 978-606-737-633-3**

Adrian PECULEA
Sorin BUZURA

Bogdan IANCU
Vlad RAȚIU

Coordinators: Vasile Teodor DĂDĂRLAT
Emil CEBUC

COMPUTER NETWORKS

Practical activities



UTPRESS
Cluj-Napoca, 2023
ISBN 978-606-737-633-3



Editura UTPRESS
Str. Observatorului nr. 34
400775 Cluj-Napoca
Tel.: 0264-401.999
e-mail: utpress@biblio.utcluj.ro
<http://biblioteca.utcluj.ro/editura>

Director: Ing. Dan Colțea

Recenzia: Prof.dr.ing. Gheorghe Sebestyen
Conf.dr.ing. Lia-Anca Hangan

Pregătire format electronic on-line: Gabriela Groza

Copyright © 2023 Editura UTPRESS

Reproducerea integrală sau parțială a textului sau ilustrațiilor din această carte este posibilă numai cu acordul prealabil scris al editurii UTPRESS.

Tiparul executat la Tipografia UTCN.

ISBN 978-606-737-633-3

Bun de tipar: 28.04.2023

Introduction

Designed as an operational tool - support for (self) training activities, the book „Computer networks. Practical activities” aims to address a wide spectrum of problems and theoretical approaches, accompanied by real examples and practical applications based on the theoretical part.

The book is addressed primarily to students following study programs at the Faculty of Automation and Computer Science, who are in their first contact with computer networks. At the same time, the issues addressed in the book, the theoretical content and practical exercises can serve as an invitation to all those interested in the study of computer networks used mainly in modern systems (teachers, researchers, students from other study fields, graduates, engineers of different specializations, etc.). The study material offers support to both students in individual and group study, orienting them towards efficient self-organization of their own activity, as well as to teachers in the optimization of the design-organization-evaluation processes, in order to ensure the quality of university training.

The primary objective of the book is to provide specific information and prepare the reader for understanding, designing, and troubleshooting computer networks. This book uses, in an operational way, the contents of the "Computer networks" course, focusing mainly on creating learning opportunities, by providing various teaching tasks, exercises, analysis, reflections, questions and comments.

The topics are designed in an active and interactive way and include essential theoretical elements, approaches to conceptual clarifications and classifications, completed by applications and tasks. The structure of the book is gradual in complexity. The practical tasks are not so much an end in themselves, but occasions, means of orientation towards the exercise of the abilities, the capacities that the students will use later, as an indicator of their professionalization in engineering.

The first part presents the main cabled transmission media used in modern computer networks and the necessary tools and techniques used to analyzed and evaluated the correct functioning of computer networks. The network layer and its protocols, together with static routing strategies, are covered in the second part of the book. Network programming aspects are introduced in the third part, mainly software for socket applications and debugging network applications. The fourth part of the book presents aspects related to the organization of local and virtual networks. The final section of the book focuses on understanding and analyzing common security threats that occur in computer networks.

We hope that this book will help in developing the specific way of thinking in the field of engineering, will expand the spirit of teamwork between students and will streamline communication, contributing to increasing the quality of university education.

The authors,

Cluj-Napoca, 2023

Content

CHAPTER 1: INTRODUCTION TO WIRESHARK AND PACKET TRACER.....	4
CHAPTER 2: COPPER BASED TRANSMISSION MEDIA AND UTP CABLING	12
CHAPTER 3: OPTICAL FIBERS AND COMPONENTS.....	20
CHAPTER 4: STRUCTURED CABLING	28
CHAPTER 6: NETWORK LAYER – IPv4 ROUTING AND DHCP	46
CHAPTER 7: NETWORK LAYER – IPv6	56
CHAPTER 8: APPLICATION LAYER: NETWORK PROGRAMMING WITH SOCKETS.....	69
CHAPTER 9: ETHERNET, ARP AND NDP	78
CHAPTER 10: VLANs, TRUNKING AND INTER-VLAN ROUTING.....	92
CHAPTER 11: LAYER 2 NETWORKS, SPANNING TREE PROTOCOL, LINK AGGREGATION AND ETHERCHANNEL.....	102
CHAPTER 12: SECURITY THREATS IN COMPUTER NETWORKS	120

CHAPTER 1: INTRODUCTION TO WIRESHARK AND PACKET TRACER

1. Objectives

The objectives of this chapter encompass three aspects:

- A brief continuation of the theoretical introduction to communication/networks and network stacks
- Introduction to Wireshark
- Introduction to Cisco Packet Tracer

2. Theoretical considerations

Please read the following before continuing. The notions presented in the practical activities are intended for use in strictly ethical and legal ways. Any other use of data derived from the information presented henceforth may be subject to the furthest prosecution of the law and in continuing these practical activities and using the information presented the students acknowledge that the Technical University of Cluj-Napoca and the staff involved are in no way liable concerning any illegal action undertaken by the entities with access to the materials presented in this book. Please use all the knowledge you are about to acquire in ethical and legal ways.

2.1 Communication/Networking

In order to successfully communicate, devices need rules. Generally speaking, devices fall into one of two categories: endpoint devices and network devices. Endpoint devices represent the communicating entities, whilst network devices represent the necessary infrastructure devices which are required for communication (e.g. houses and postal services; physical locations and roads, signs & regulations).

2.2 Network Stacks

Network stacks are an essential concept to networking. Please consider the important terms and differences between them. A stack model represents a separation of functions (which is subject to IEEE regulations and directives) when handling networking. An actual stack represents the exact combination of protocols implemented at each layer and their specific configuration (e.g. a decent analogy: a breakfast might be milk and cereal as a stack model, but as a stack it might be organic 3.5% fat milk and buckwheat cereal). The contents of this practical activity will mainly work with the TCP/IP stack model (Figure 1.1), due to the fact that the Internet was designed with the TCP/IP model in mind, before the more refined OSI model was adopted. Note: there are many efforts to migrate the Internet towards the OSI model and many more different networks which use the OSI model (e.g. industrial networks or IoT networks).

Important note: Messages are referred to as follows:

- PHY – bits/symbols
- DLL – frames
- Internet – packets

The practical activity contents only investigate wired media, although some wireless aspects will be briefly presented in the future.

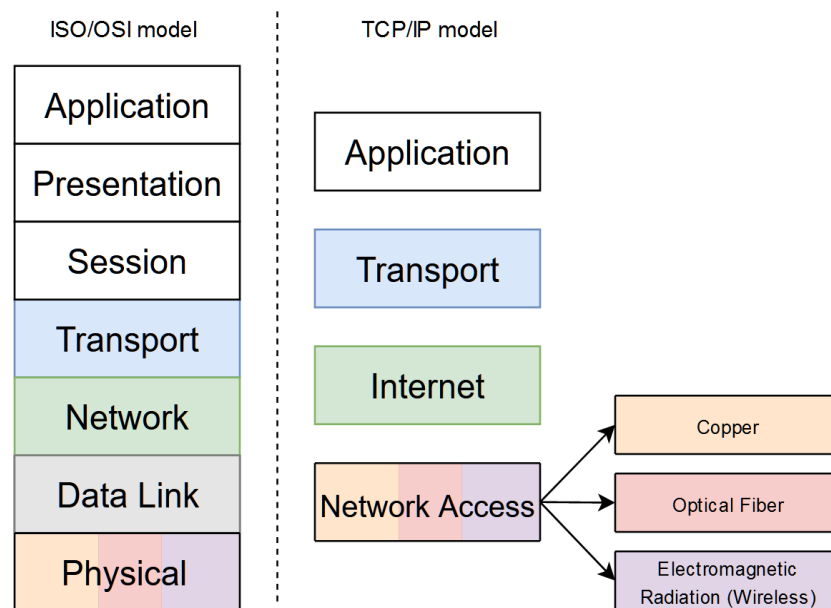


Figure 1.1 Network stack models

3. Practical activity

3.1 Install and verify Wireshark functionality

Wireshark is a packet analyzer (sometimes called a “sniffer”). In order to capture network traffic, Wireshark needs a specific interface on which to capture traffic. Please note that, even though the exercises in this section consist of investigating local traffic, Wireshark can be used to identify traffic which is neither generated locally, nor destined to the local host, depending on the interface used or logs captured through other tools. If you are working on your own workstation, navigate to www.wireshark.org and install Wireshark.

You will investigate Wireshark captured packets, especially correlating data with the corresponding TCP/IP stack layer. You don’t need to understand this data now, but it is necessary that you understand that a packet contains data which is always correlated to one of the stack layers – learning to identify the layers will be extremely useful in your future career as an engineer (and necessary for this course/practical activities).

In order to launch your first Wireshark capture, open Wireshark and select the interface in use. This is probably your LAN connection or, as seen in Figure 1.2, the Wireless Network

connection (note: Wireshark can also sniff Bluetooth and USB interfaces, among others, which are all beyond the scope of this course/practical activity).

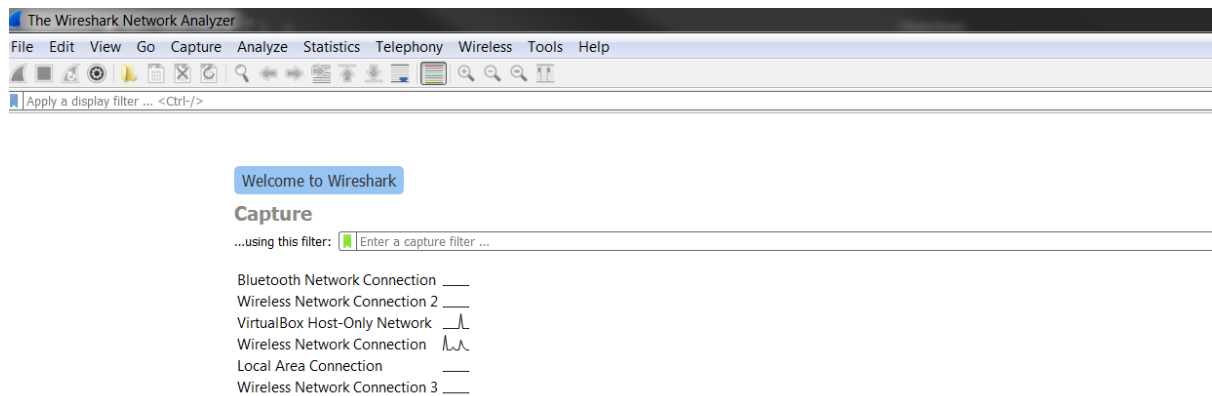


Figure 1.2 *Wireshark interfaces*

You can either double click the desired interface or navigate to Capture -> Options -> Start, as seen in Figure 1.3.

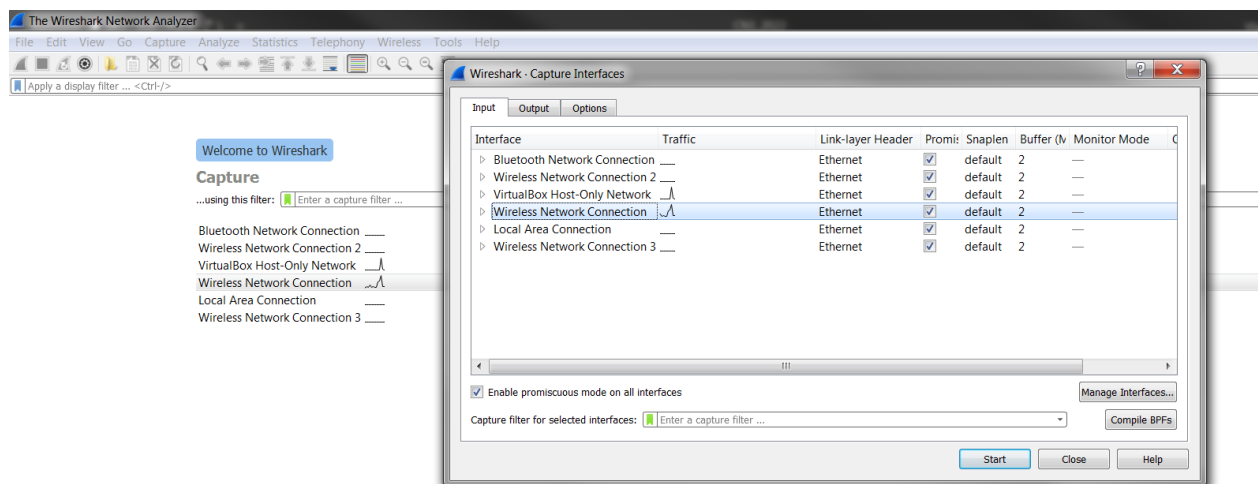


Figure 1.3 *Select Wireshark interfaces*

You should start seeing captured packets (Figure 1.4). Once you feel that you have captured enough packets go ahead and stop the capture from the indicated selection (or don't, but your local memory won't be happy).

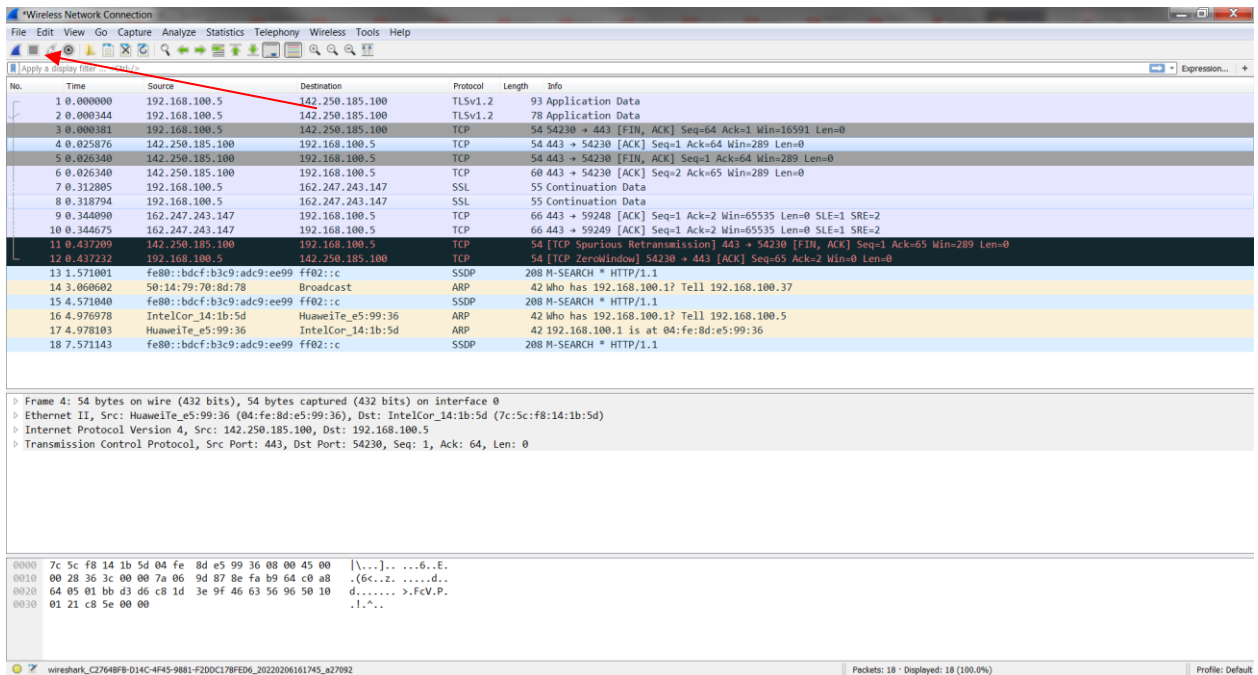


Figure 1.4 Wireshark packet analysis

You will probably see a great deal of packets, feel free to explore them, but focus on the second view (Figure 1.5), which associates data to each TCP/IP stack layer. It is extremely important to be able to associate the data in this view with each individual layer. The first four layers of the network stack are presented in this view as follow, from top to bottom: Physical, DLL, Internet, Transport. Notice that each layer presents the specific running protocol.

```

▶ Frame 5: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 0
▶ Ethernet II, Src: IntelCor_14:1b:5d (7c:5c:f8:14:1b:5d), Dst: HuaweiTe_e5:99:36 (04:fe:8d:e5:99:36)
▶ Internet Protocol Version 4, Src: 192.168.100.5, Dst: 40.101.55.130
▶ Transmission Control Protocol, Src Port: 57631, Dst Port: 443, Seq: 1, Ack: 86, Len: 0
  
```

Figure 1.5 Individual packet contents

You can view the detailed contents of each packet by clicking the drop-down arrow next to each stack layer, as seen in Figure 1.6.

```

Transmission Control Protocol, Src Port: 60853, Dst Port: 80, Seq: 1, Ack: 1, Len: 653
  Source Port: 60853
  Destination Port: 80
  [Stream index: 4]
  [TCP Segment Len: 653]
  Sequence number: 1 (relative sequence number)
  
```

Figure 1.6 Detailed packet contents

You can go ahead and capture as many packets as you like. “Start capturing packets” quick access button is located besides the “Stop capturing packets” button. You can use this button to start capturing packets without changing the interface. Whenever you quit Wireshark or start a new capture after running a previous one, Wireshark will ask you if you want to save the

captured data somewhere. Saving data for future analysis is useful in forensically investigating network activities (including attacks) but, for all intents and purposes of this practical activity, you don't need to save this data anywhere (unless you see something interesting and want to check it out later).

Following are two challenge activities.

- Challenge activity 1: try pinging some IP addresses, using the “ping ip_address” command from the console (cmd) of your workstation. Ping is a network connectivity testing tool. The networking equivalent to “Hello World” is pinging yourself (“ping 127.0.0.1” or “ping localhost”). Pinging yourself is used to test network interface functionality (e.g. if you don't have any network hardware installed on your workstation, or if it's malfunctioned, the pings will fail). Can you see the ping command on a Wireshark capture? Is it atomic or composed of multiple messages?
- Challenge activity 2: with Wireshark capturing packets on the local interface, attempt a login connection to an HTTP website, then to an HTTPS website (they don't need to be successful). Nowadays most websites offer HTTPS services and not HTTP, but you can still find some HTTP websites by searching the internet on a search engine of your choice. Can you see the username/password in the contents of the packet on a HTTP login attempt? What about the HTTPS login? (hint: can you even see HTTPS packets?). Can you explain your findings? Which TCP/IP stack layer is responsible with handling HTTP and HTTPS? (tip: try to figure it out by yourself but search when necessary). To make this challenge easier, Wireshark provides a packet filtering mechanism which you can use either during the capture or afterwards. This will be further investigated in more detail in the next practical activities. For now, in order to filter packets either type the string filter or select more options from the indicated button (Figure 1.7) and select the “Apply this filter string to the display” button (Figure 1.8). Click the adjacent “X” in order to reset the filter.

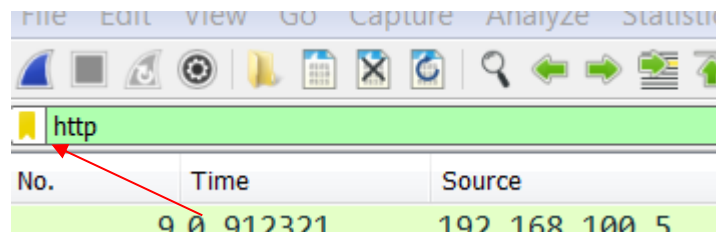


Figure 1.7 Wireshark filter

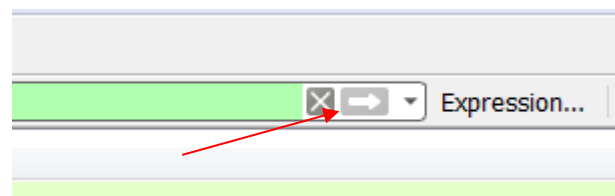


Figure 1.8 Apply filter button

3.2 Install and verify Cisco Packet Tracer

Cisco Packet Tracer (PT) is a network simulation tool provided by Cisco Systems. It is extremely useful in designing networks without/before access to physical equipment.

Please download and open the provided Intro.pkt file. You should log into PT with your Cisco Netacad/Skillsforall account. If you don't already have one from downloading PT, please create one. The file contains a previously configured network. During the course of the semester, you will learn skills that will allow you to configure and debug a similar network. Note: PT provides simulation functionalities for both wired and wireless networks; during the course of this practical activity you will focus on wired networks.

Go ahead and take a look at PT basics. The two main categories of devices you will use are network devices (which provide network infrastructure) and endpoint devices (which, generally, are computers, servers, etc.). Try and add a PC to the network (Figure 1.9) either by clicking on the PC, then on the canvas or by dragging and dropping the PC. You can erase a device from the canvas by selecting Delete, then clicking on the device, or by dragging a selection of one or more devices and clicking delete.

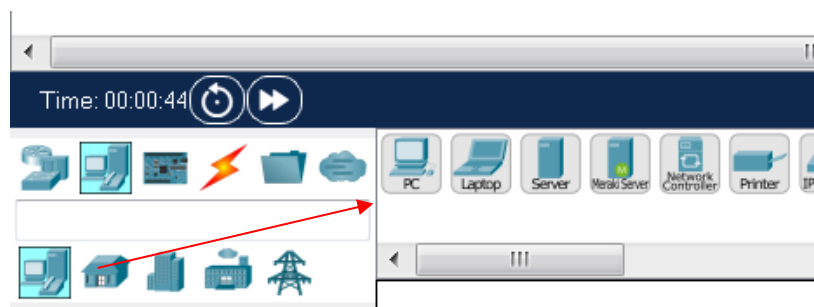


Figure 1.9 Packet Tracer section for adding network components

Connect it to the switch, using an automatic connection (Figure 1.10).

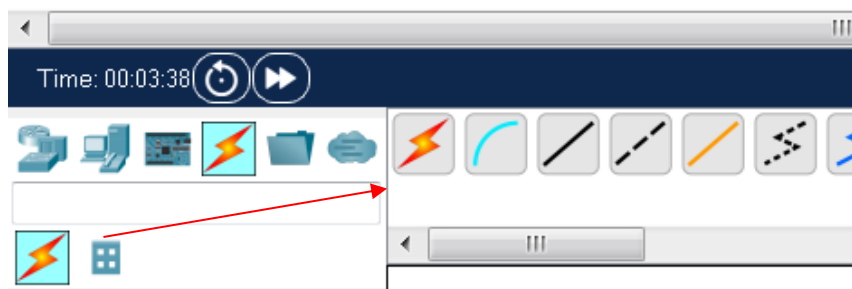


Figure 1.10 Cable button to *connect PC to switch*

Select the automatic connection option, then click on the PC and finally on the switch. Don't worry too much about understanding what is going on at this point, you will go into more detail in future activities.

In order to test connectivity, you can access any PC by clicking it; select Desktop -> Command prompt and ping an IP address on the network. You can view these IP addresses by hovering your cursor above any device. You should see something similar to the previous activity (Figure 1.11). Try pinging from your newly added PC. Do the pings work? Can you figure out why? (Hint: try comparing the configurations of your PC with the previously existing PCs). This will be further analyzed in the following practical activities.

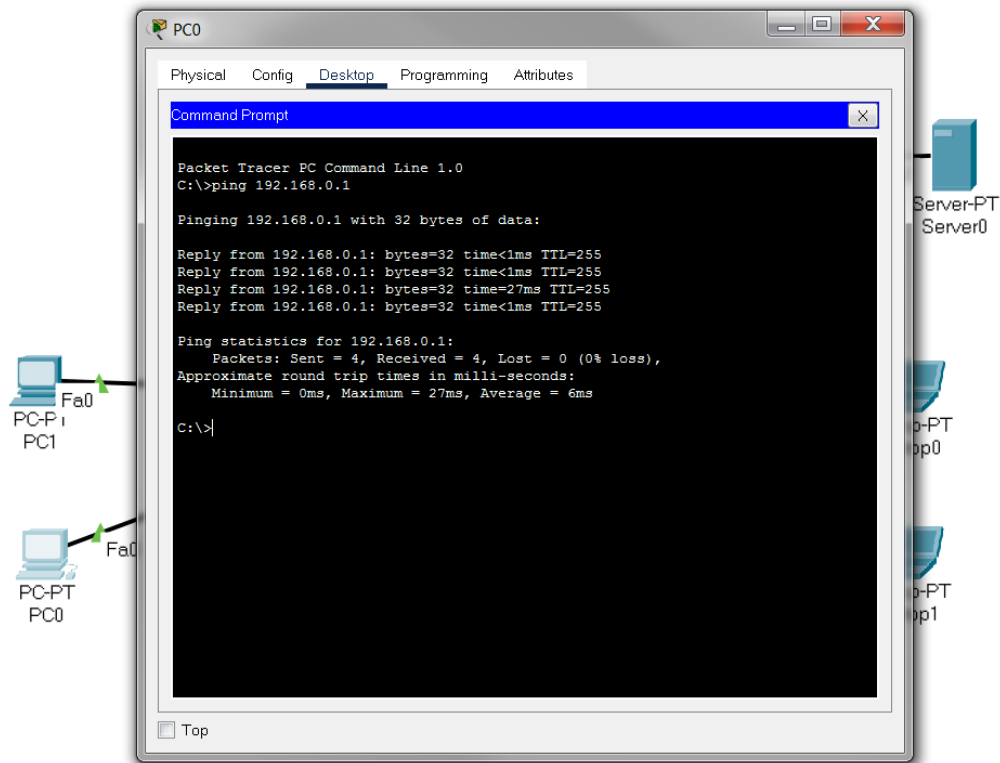


Figure 1.11 Ping in Packet Tracer

PT can run in Realtime or Simulated mode (Figure 1.12). Try switching to simulated mode. Run a ping command and press the play button. You can change the simulation speed from the slide bar. You should be able to see packets moving through the network. The exact rules and nature of the message exchanging process will be presented in future activities.

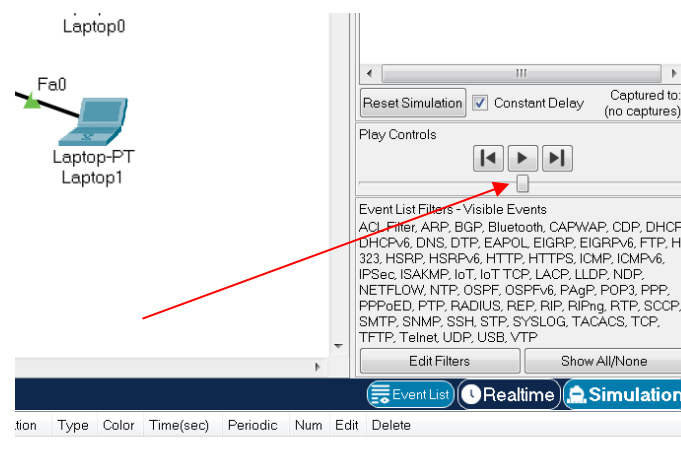


Figure 1.12 Real-time and simulation mode selection in Packet Tracer

PT has many functionalities which you are welcome to explore. The more detailed functionalities will be presented as you move on to future activities.

3.3 Questions

- What is a network stack model? What is a network stack?
- What is Wireshark and what is it used for?
- What is Cisco Packet Tracer and what is it used for?

CHAPTER 2: COPPER BASED TRANSMISSION MEDIA AND UTP CABLING

1. Objectives

The objective of this work is knowledge and understanding of copper-based transmission media, the main associated parameters, as well as the wiring and testing of UTP cabling.

2. Theoretical considerations

ISO Open Systems Interconnection (OSI) reference model (Figure 2.1) incorporates 7 layers (Physical, Data Link, Network, Transport, Session, Presentation and Application). The first layer defines the physical/hardware concepts of a communication network; the first three layers use physical/hardware components. The remaining four layers define the logical concepts of a communication network. The current practical work focuses on the Physical layer of the ISO/OSI stack (or the Network Access layer of the TCP/IP stack), mainly copper-based transmission media.

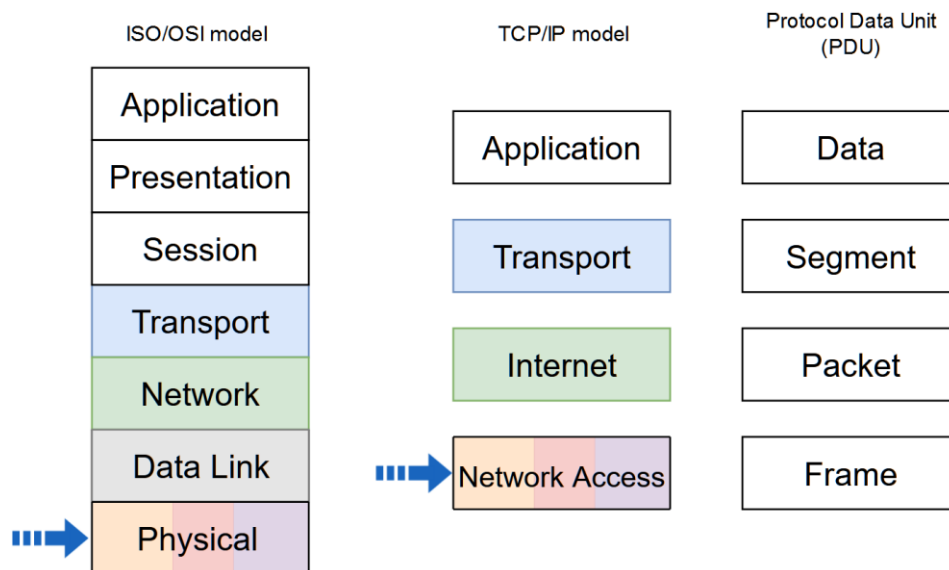


Figure 2.1 Network stack models and PDU naming in each level. The arrows indicate the addressed layers in the current activity

In Europe, the ISO/IEC-11801 standards family defines general and specific cabling design documents. It comprises the ISO/IEC 11801-1:2017 Information technology — Generic cabling for customer premises — Part 1: General requirements and includes ISO/IEC 11801-2, ISO/IEC 11801-3, ISO/IEC 11801-4, ISO/IEC 11801-5, ISO/IEC 11801-6. The ISO/IEC 11801-1 specifies the requirements for coaxial, twisted-pair copper and optical fiber. In the USA and Canada, ANSI/TIA-568-C standard is used instead of ISO/IEC 11801.

2.1 Coaxial and twisted cables

In data transmission, transmission media represents the physical path between the transmitter and the receiver; it must guarantee superior performances expressed as parameters like communication speed, transmission error rate, cost, amplification requirements.

Data transmission characteristics and quality are determined both by the transmission support media and propagated signal's characteristics. IEC 61935-1 standard is used for "reference measurement procedures for cabling parameters and the requirements for field tester accuracy to measure cabling parameters".

In data transmission systems design some determinant elements for system performance are:

- *bandwidth* – represents the transferred data volume on a communication channel so that if the other factors remain constant, the larger the bandwidth the better signal transmission rate will be obtained;
- *interference* – is generated by signals superposition in the same frequency band, fact that can generate signal distortion. Correct shielding of the transmission media can determine the minimization of this type of effect;
- *number of receivers* – assumes point to point or shared links construction.

The main electric parameters of the copper-based transmission media are:

- *impedance* – for data transmission is important not only the impedance value at a given frequency but also its variation function of frequency;
- *propagation speed* - represents a percent from the light speed;
- *attenuation (insertion loss)* – the channel behavior at frequencies depends on this parameter. This value increases in proportion with the cable length;
- *crosstalk* – is the measure of the influence produced by a cable to another cable placed in its vicinity.

Coaxial cable is a versatile transmission media, used in a large application variety, from long distance telephonic transmission, local area networks, to TV distribution for various devices connection. This is a media that allows the operation on a large frequency spectrum.

The cable contains a copper core insulated from the second exterior conductor, made as a shield from a thin wire braid (Figure 2.2).

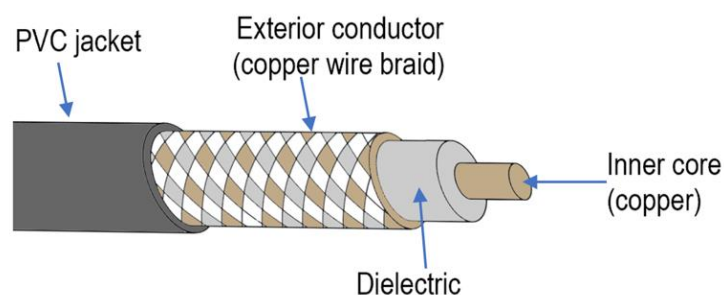


Figure 2.2 Coaxial cable structure

The main characteristics of the coaxial cable are:

- allows digital and analog signal transmission;
- because of concentric construction type, is resistant to magnetic interference.

Modern Internet topologies use optical fiber to transport data from an ISP (Internet Service Provider) to local communities and, from there, they use coaxial cable (available from CATV) to connect the subscribers. At the subscribers' end a cable modem acts as a bridge between the coaxial network and the customer LAN. For achieving high-bandwidth data transfers standard specifications, such as Data Over Cable Service Interface Specification (DOCSIS), are used for hybrid fiber coaxial (HFC) networks (for example, DOCSIS 3.1 specifies a downstream data speed of 10 Gbps and upstream speeds of up to 1 Gbps). Modern distribution systems are now also using Ethernet Passive Optical Networks (EPON) Protocol over coaxial media (EPoC) (IEEE Std 802.3bn) with a physical layer specification for up to 10 Gb/s downstream and up to 1.6 Gb/s upstream point-to-multipoint link. Other applications, such as 10Gb Ethernet up to 100 Gb Ethernet for full-duplex point-to-point links between network devices, use a special coaxial cable with two inner conductors called Twinaxial cabling or Twinax.

For LANs this cable type has been replaced with other high-bandwidth technologies because its performances were reached and exceeded for short distances by twisted cable and for long distances by optical fiber.

The main constraints related to performances refer to its attenuation, inter-modular noises and heating.

Coaxial cable used in local area networks had 50 Ohms impedance and were of 2 types:

- *thin coaxial cable* (RG58 in **10BASE2** type IEEE 802.3 networks) is the most widespread and used for interior installations because of a good price/performance ratio;
- *thick coaxial cable* (RG213 in **10BASE5** type IEEE 802.3 networks) is used for exterior installations because of a higher mechanical resistance and a better length limit.

Computers connection to coaxial cable was done using two methods: using T junctions or special connectors called *vampire tap connector* placed into a device called transceiver, which allow their thrust in cable without its cutting necessity. The connector penetrates the insulator layer making the contact directly with the conductor layer. The connection between transceiver and network interface card is done with a transceiver cable that is connected to AUI port (Attachment Unit Interface). For long distances analog transmission, signal amplifiers are required, and for digital signals, repeaters are required, the standards specifying exactly their placement distance. In thin coaxial cable case, the maximum distance was 185m and in thick coaxial cable case, the maximum distance was 500m. For Twinax, SFP+ (SFP = small form-factor pluggable) network interface module can be used.

Twisted cable (*twisted pair - TP*) or cable with twisted pairs of copper wires, having a common outer jacket (with or without shielding), represents the usual cable type used in local area networks and telephone system. The purpose of wire twisting is the reduction of magnetic distortion, of interferences between adjacent pairs of cable. This cable acts like a single communication link. For cables with several twisted wire pairs, twisting steps must be different for each pair so that the crosstalk between pairs to be minimum. Because of the progress realized in TP cable manufacturing technology, these can be used in a very large frequency

range allowing Gbps data transmissions, and in Gigabit networks offering for short distances performances comparable with optical fiber. TP cable represents the transmission media for analog and digital signals usually used in telephone system and local area networks.

TP cables used in computer networks have four pairs of twisted wire, allow a maximum distance of 100m and are being used in 10, 100, 1000Mbps and 1, 2.5, 5 or 10Gbps networks. The standard cable impedance is 100 Ω . Furthermore, 25 and 40 Gbps networks allow a maximum distance of 30 m using Class I and II cables (cat. 8.1 and 8.2). Different diameter for the copper cables exist and are measured using AWG standard (American Wire Gauge): from 22 AWG to 26 AWG and, for short distance also 28 AWG.

Twisted cables categories used in data transmissions are differentiated function of supported utilizations. Table 2.1 presents balanced (symmetrical) Twisted-Pair class specifications.

Table 2.1 *Balanced Twisted-Pair Class Specifications*

Class	Bandwidth	Category
Class A	up to 100 kHz	Category1
Class B	up to 1 MHz	Category2
Class C	up to 16 MHz	Category3
Class D	up to 100 MHz	Category5e
Class E	up to 250 MHz	Category6
Class EA	up to 500 MHz	Category6a
Class F	up to 600 MHz	Category7
Class FA	up to 1000 MHz	Category7a
Class I and Class II	up to 2000 MHz	Category8.1, 8.2

All these classifications do not refer only to cables but also the entire associated connecting system: connectors, outlets, patch panels etc. Twisted cable allows point to point connection implementations, realizing different star or extended star network topologies.

The cable naming convention (Table 2.2) from ISO/IEC 11801 presents the different types of cable construction, based on their screening: XX / XXX. Examples of cable naming are: U/UTP, U/FTP, F/UTP, S/UTP, SF/UTP, F/FTP, S/FTP, SF/FTP etc.

Table 2.2 *Cable naming*

XX				/	X	XX	
overall screen					element screen	balanced element	
U = unscreened	F = foil screened	S = braid screen	SF =braid and foil screen		U = unscreened	F = foil screened	TP

In Figure 2.3, F/UTP and U/FTP cables are presented.

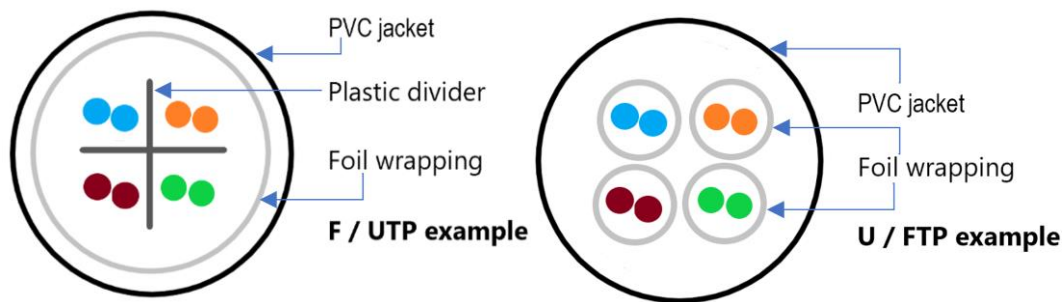


Figure 2.3 F/UTP and U/FTP

2.2 UTP cabling

At Ethernet and Fast Ethernet networks UTP cabling 1-2 wire pair is used for transmission and 3-6 pair for reception. This wire type arrangement is called MDI (Media Dependent Interface) or normal arrangement. Usually, the wires are connected according to the same rule in the connectors at the two ends of the cable, case in which the cable (patch cord) is called *straight-through*. The straight-through cable was designed to be used when connecting two devices of different type (ex. computer – modem, router – switch etc.)

In some special cases the reception must be reversed with transmission in order to enable communication, case in which the cable (patch cord) is called *crossover*. As specified by IEEE 802.3 the crossover function connects the transmitters of one end to the receivers the other end of the link segment. The crossover cable was designed to be used when connecting two devices of the same type (ex. computer – computer, router – router etc.). In modern interfaces, the *Automatic MDI/MDI-X* function automatically detects the needed cable connection type and configures the correct connection, thus *straight-through* cable can be used throughout the network.

Table 2.3 EIA/TIA-T568-A

Pin#	Pair#	Function	Wire color	Used with 10/100BASE-T	Used with 1000 BASE-TX
1	3	BI_DA+ (Transmission+)	White/Green	Yes	Yes
2	3	BI_DA- (Transmission-)	Green	Yes	Yes
3	2	BI_DB+ (Reception+)	White/Orange	Yes	Yes
4	1	BI_DC+	Blue	No	Yes
5	1	BI_DC-	White/Blue	No	Yes
6	2	BI_DB- (Reception-)	Orange	Yes	Yes
7	4	BI_DD+	White/Brown	No	Yes
8	4	BI_DD-	Brown	No	Yes

At Gigabit Ethernet networks UTP cabling all four wire pairs are used both for transmission and reception. UTP cables contain four twisted wire pairs each pair being identified through a color: blue, orange, green and brown. Each pair contains a colored wire and a white combined with the respective color wire. The connectors used for this cable are RJ-45 type male connectors containing 8 pins corresponding to the 8 wires. Viewed from the front, the pins are numbered from 1 at right to 8 at left. The wire connection mode to the pins determines the cable type. There are two standards for wire connection to RJ-45 connector: EIA/TIA-T568-A

and EIA/TIA-T568-B. These connections are presented in Table 2.3 - 2.5 (BI_DX means Bi-directional pair X).

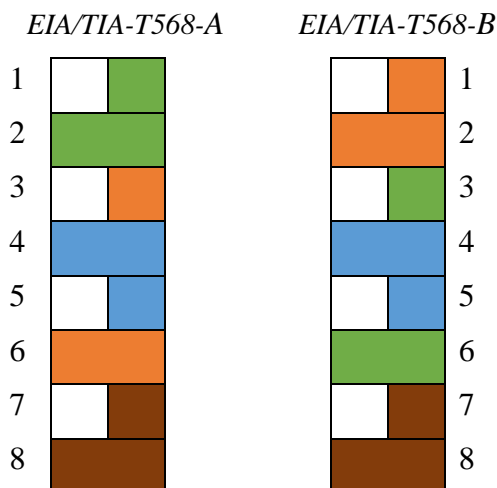
Table 2.4 EIA/TIA-T568-B

Pin#	Pair#	Function	Wire color	Used with 10/100BASE-T	Used with 1000 BASE-TX
1	2	BI_DA+ (Transmission+)	White/Orange	Yes	Yes
2	2	BI_DA- (Transmission-)	Orange	Yes	Yes
3	3	BI_DB+ (Reception+)	White/Green	Yes	Yes
4	1	BI_DC+	Blue	No	Yes
5	1	BI_DC-	White/Blue	No	Yes
6	3	BI_DB- (Reception-)	Green	Yes	Yes
7	4	BI_DD+	White/Brown	No	Yes
8	4	BI_DD-	Brown	No	Yes

Thus, to obtain a straight-through cable both ends of the cable must be connected according to the same standard (A-A or B-B) and in order to obtain a crossover cable each end of the cable must be connected according to a different standard (A-B or B-A).

Table 2.5 a. Color coding example

b. RJ-45 connectors



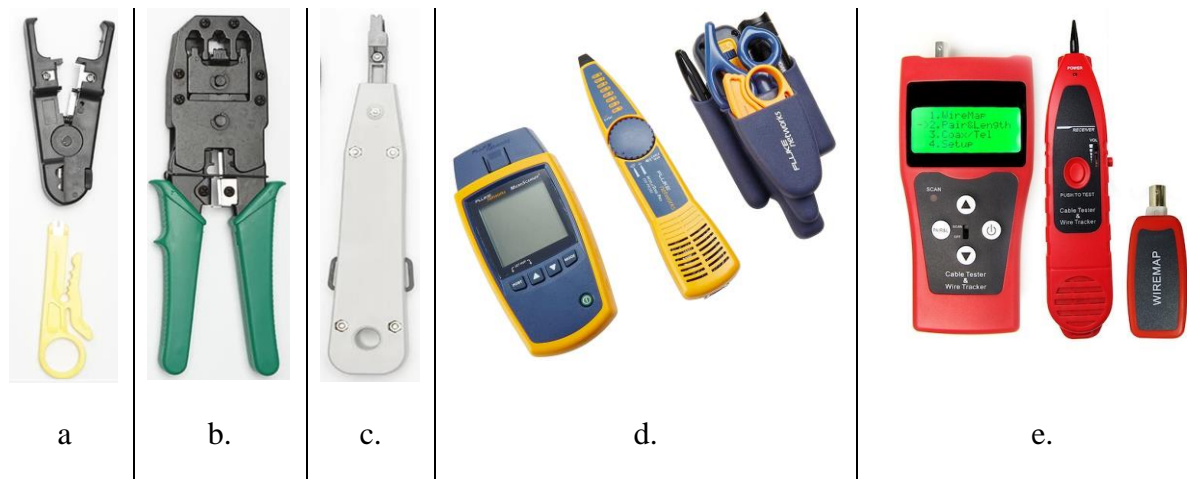
At UTP cabling are used both passive and active devices. Passive devices are not supplied from a voltage source while active devices require power supply. The most important passive devices are: RJ-45 connector (8P8C connector), outlet and patch panel. At layer 1, the most important active device is the transceiver. An RJ-45 connector is a device with eight pins in which a TP cable is connected. It enters in outlets and patch-panels structure. In outlets are connected the computers using patch cords. The outlets are connected to the patch panel which is located in the distribution closet. Using a patch cord, the patch panel is connected to the switch which is also located in the distribution closet. The switch is a multiport bridge. The transceiver is a bidirectional device which receives the signals from one type of interface, converts them in specific signals for another type of interface and transmits them to an interface of that type.

Based on IEC 61935-1 standard, tests are performed to measure the cabling parameters: insertion loss, propagation delay and delay skew, near-end crosstalk (NEXT) and power sum

NEXT, far-end crosstalk (FEXT) and power sum FEXT, different attenuation and crosstalk types.

For cabling a TP cable and a TP outlet, several tools should be used, as shown in Table 2.6: cable stripping/cutting tool (a.), cable crimping tool (b.), punch down tool (c.) and cable testers (d., e.).

Table 2.6 *Cabling and testing tools*



On the market, both flat or round twisted pair cables can be found (Figure 2.4). They serve the same purpose, but they can be used in different scenarios. Round cables are the most often used cables in networking, however flat cables could be useful to run under a carpet, along a wall or a corner.



Figure 2.4 *Flat cat.6 cable (left) and round cat.6 cable (right)*

3. Practical activity

3.1 UTP cable connection and testing

- Identify the wired NIC (network interface card) on your lab computer.
- Identify the cables used in the laboratory (PC – outlet). What standard was used for cabling them?

- Using the EIA/TIA-T568-A or B standard, straight-through cables will be made and tested.
- Using the EIA/TIA-T568-A and B standard, crossover cables will be made and tested.
- Using the punch down tool cable a wall outlet.
- Research online the different types of cable discussed in the practical work.

3.2 Network cabling and testing

- The connectivity between two computers will be tested using crossover cables.
- Cable and test the connectivity using the network presented in Figure 2.5 (using PCs, a patch panel and a switch).

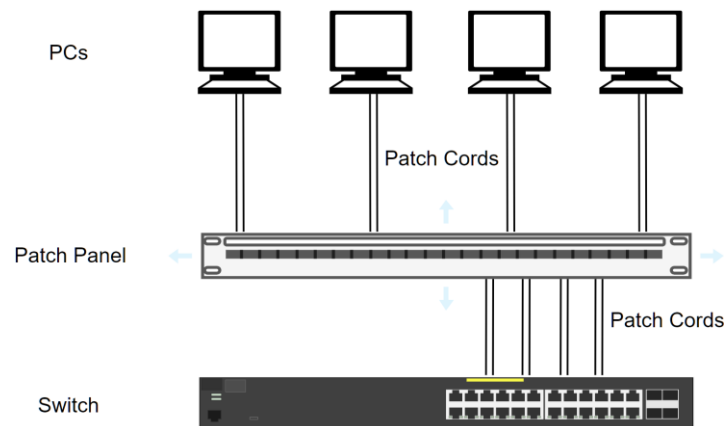


Figure 2.5 Cabling test network

3.3 Questions

- Why are screening and shielding important in choosing the appropriate network cable?
- Why TP cables used in computer networks are limited to a maximum distance of 100m?
- Which type/category of cable should be used for a new LAN?

CHAPTER 3: OPTICAL FIBERS AND COMPONENTS

1. Objectives

The objective of this work is to gain knowledge on optical fibers and components, link performance analysis and the optical power budget calculus.

2. Theoretical considerations

2.1 Optical fibers and components

The current practical work continues the focus on the Physical layer of the ISO/OSI stack (Figure 3.1) by providing knowledge on optical fibers and components.

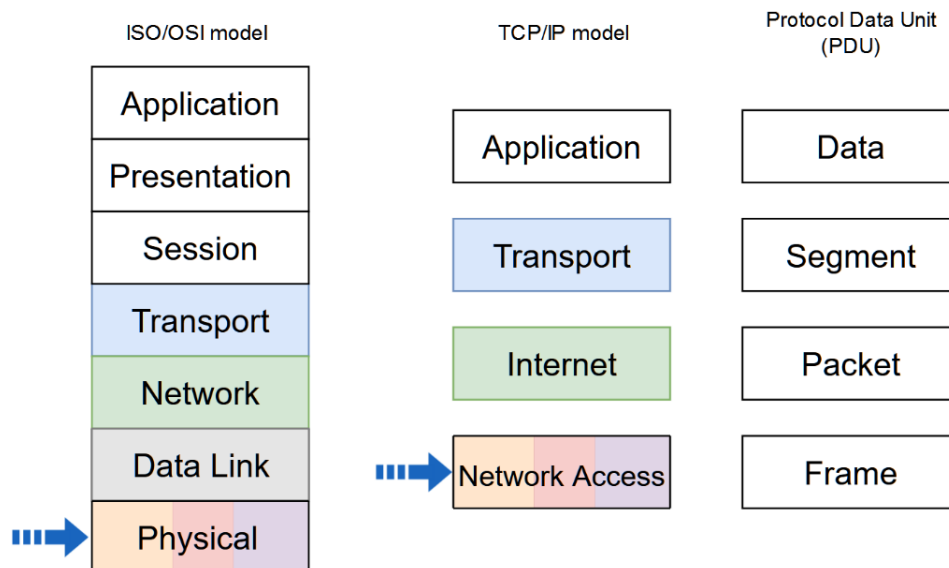


Figure 3.1 Network stack models and PDU naming in each level. The arrows indicate the addressed layers in the current activity

Once the drop in the price of optical fibers, and appropriate communications equipment, this has become the environment of choice for new high-speed connections (exterior and interior).

To transmit data, optical fibers send light signals along glass or plastic cores (of the order tens of microns (μ), which constitutes a wavelength guide for light, obtained from a combination of silicon dioxide and other elements).

An optical *fiber strand* is the basic element of an optical fiber cable (a cable contains several strands). A strand has three layers: core, cladding and coating. A fiber optic cable consists of several components: fiber strand(s), buffer, protective materials, outer jacket.

The core is wrapped by material made of silicon dioxide having a refractive index lower than the core called cladding. In order to protect the cladding, this is wrapped in a plastic material. This is called buffer and is wrapped in a material, usually Kevlar, which confers resistance of fiber at the time of installation. Optical fiber buffers are of two categories: tight (a protective covering is applied over the coating of each fiber strand) or loose-tube (several strands inside a tube filled with a protective gel). For outdoor, long-distance installation, loose-tube fiber is preferred. The last wrapper is the jacket which protects the fiber against abrasive materials, solvents and other factors. The color enclosure in the case of multimode optical fiber is usually orange and in the case of single-mode optical fiber is usually yellow. Each fiber optics cable is composed of two fibers wrapped separately, a fiber being used for transmission and another for the reception, ensuring in this way a full-duplex connection. A cable of optical fiber may contain from two up to hundred separate fiber strands (usually in LANs, up to 24). Figure 3.2 presents the layer of an optical fiber and an optical fiber transversal section.

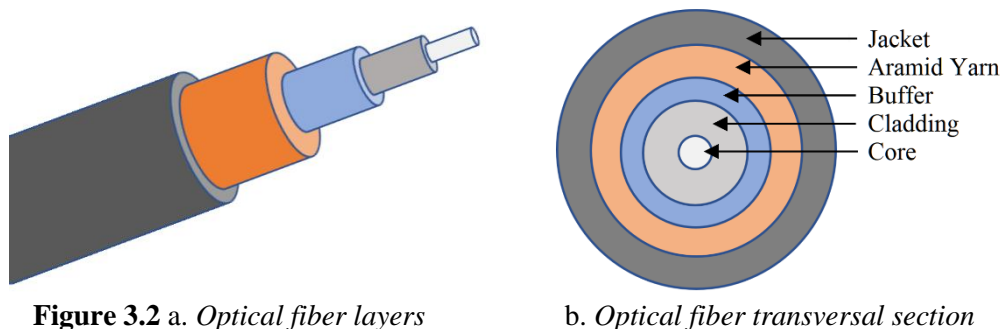


Figure 3.2 a. *Optical fiber layers*

b. *Optical fiber transversal section*

For the signal to be reflected without loss, the following two conditions need to be met:

- Optical-fiber must have a refractive index higher than the material surrounding it;
- The angle of incidence of light signal must be greater than the critical angle of fiber and of the material surrounding it. The angle of incidence of light signal can be controlled by using the next two factors:
 - Numerical aperture of the fiber is the range of angles of the light signal for which the reflection is complete;
 - The modes are the ways that the signal light can follow.

Unlike copper-based transmission media, optical fiber is not susceptible to, and it does not generate electromagnetic or crosstalk interference.

Two main optical fibers are commonly used in LANs and WANs: *single-mode* and *multimode*. Single-mode optical fiber is used for long distance links and for vertical cabling in buildings (building's backbone). Multimode optical fiber is commonly used in horizontal and vertical cabling. Multimode fiber has a larger core diameter compared to single-mode. Thus, multimode does not require the same precision as single-mode, resulting in less expensive connectors, transmitters etc.

For the **single-mode fiber** the core diameter is small enough as to permit only one mode (one way) light signal, being sent in a straight line through the middle of the core. Single-mode optical fiber cables use cores with diameter between 8μ and 10μ . The most used single-mode optical fibers have 9μ diameter and cladding with a diameter of 125μ . They are usually referred as $9/125\mu$ optical fibers. Light source used is the infrared laser. It is recommended caution when using lasers as source of light since it may affect the eyes. Single-mode fibers may transmit data at distances over 100km. The loss on km of single-mode optical fiber is specified

by the manufacturer. In the case of single-mode fiber, the refractive index of glass stays constant. This type of glass is called step index glass.

The core of **multimode fiber** has a sufficiently large diameter as to permit several modes (several ways) for light signal. Standard multimode optical fiber cables have a core diameter of 62, 5 μ or 50 μ and cladding with a diameter of 125 μ . They are usually referred as optical fibers of 62.5/125 μ or 50/125 μ . Usually, the light sources used with multimode fibers are Infrared Light Emitting Diode (LED) or Vertical Cavity Surface Emitting Lasers (VCSEL). LED-s are cheaper and require less safety measures than lasers. The disadvantage of LED is that may not transmit light signals at distances as large as lasers. Multimode fibers of 62.5/125 may transmit data at distances of up to 2000m. The loss of multimode optical fiber is specified by the manufacturer. In the case of multimode fiber, the refractive index of glass may be constant (multimode step index glass) or may also decreases from the center to the exterior (variable or graded-index glass and allows various illuminating modes to reach the receiver at the same time).

In optical fiber, beside propagation, the light is subjected to two main phenomena: attenuation and dispersion. Attenuation or absorption is essentially due to the presence of hydroxyl ions - OH and of the various metal ions. Light may also be spread by micro crystals, lower than the wavelength, which form at the cooling of the glass. Attenuation limits the length of optical fiber to be used. The dispersion or impulse width widening is mainly due in multimode fibers to the different length of the modes. The chromatic dispersion appears due to the variation of the refraction index function of the light color or wavelength. The dispersion limits the use of optical fiber in the frequency or in bandwidth. The two limitations multiplied characterize most accurate an optical fiber. 20MHz-km values are obtained for fiber with step index, 1GHz-km for the variable index and 1000GHz-km for the single-mode in which there is no modal dispersion.

Optical fiber **transmitters** convert electrical signals in equivalent luminous pulses. There are two types of light source used by transmitters for optical fiber:

- The LED which produces infra-red light having a wavelength of 850nm or 1310nm. They are used with multimode fibers. Coupling to optical fiber can be improved by using a spherical lens;
- LASER semiconductor diode containing which produces infra-red light having a wavelength of 1310nm or 1550nm. They are used with multimode or single-mode fibers.

There are two types of basic design for LEDs: with surface emission and with edge emission. At surface emission led, the emission of light is perpendicular to the plane of junction through a thin transparent layer. They emit in a geometric radial spectrum. At edge emission led the light is emitted in a plane parallel to the junction at semiconductor edge. The materials used are often compounds III V as GaAs or Al_xGa_{1-x}As for wavelengths of 0.8-0.9 μ m and Ga_xIn_{1-x}P_yAs_{1-y} for wavelengths of 1.3-1.6 μ m. Emission spectrum of a LED is between 25 to 40 μ m for small wavelengths and 50-100 μ m for larger wavelengths.

LASER semiconductor diodes, laser diodes (LD), are obtained by introducing a led into an optical resonant cavity. The effect of laser only appears at the existence of a direct current high enough to achieve an inversion of the population of the electrons and holes from the two energy strips of conduction and valence. The current value from which this effect appears is called limit current. Under this current the device acts as an ordinary led. Since the light emitted by a

laser is much more coherent than issued by a LED, the efficiency of the optical fiber coupling is higher. Optical power also captured by laser is greater than that emitted by the LED.

An analysis compared between the two types of transmitters is clearly in favour of LD because the possibility to use higher frequencies, narrower spectrum and in favour of the LED due to price and power stability in relation to temperature. The life expectancy of both devices is equal and is of the order of 10 million hours.

The fiber optics **receivers** convert luminous pulses into equivalent electrical signals. Semiconductor devices normally used for optical fiber are classified in two types: simple and with internal gain. The first may be called PIN photodiode by type of doping (p intrinsic and n) and the second category is called APD (Avalanche Photo- Diodes). These devices are sensible at 850, 1310 and 1550nm wavelengths, wavelengths used by transmitters for optical fiber. As semiconductor materials are used Si for wavelengths of 800-900 nm and Ge or InGaAsP for 1300 and 1500 nm. Si has optimum sensitivity only within a reduced frequencies range but Ge has an appreciable darkness current and is more sensitive to noise. For this reason last possibility is the best but requires a more sophisticated manufacturing technology and therefore has a higher price.

In order to connect multiple fibers or for achieving a longer fiber, **splices** (junctions) may be used. Splices are of two types: mechanical and fusion. Attenuations introduced are lower than 0.5dB (ANSI/TIA-568-C.3 specifies that mechanical or fusions splices shall not exceed a maximum optical insertion loss of 0.3dB). At mechanical splices the two ends of the fiber, carefully cut, cleaned and polished are caught in a rigid mechanical holder that they fix to each other in an fixed ensemble. Fusion splices shall be carried out by heating close to the melting point. At this moment the two fibers are pressed against one another and cooled. These operations shall be preceded by cutting operations and finishing their ends and prior alignment of the two ends which will be connected. Fusion splices also remake draw/bursting resistance of the fiber at approximate 90% of the original value. To protect the splices, splice enclosures are used.

Connectors in the optical fiber allow the connection to ports. The common used connectors are SC (Subscriber Connector) - snap on type, ST (Straight Tip) - twist on type, FC (Ferrule Connector) - screw on type, LC (Lucent Connector) - snap on type and MTP/MPO - push/pull type, for multimode optical fibers and for single-mode optical fibers. Attenuation introduced by an optical connector, even of superior quality is greater than that introduced by a splice, having values of approximately 1 dB. Connectors are high precision mechanical equipment and usually one end of the fiber is in the connector and one is free. In this case attaching a connector shall be reduced to the execution of a splice. Such a solution is usually more advantageous than mounting a connector directly to the end of the fiber because prefabricated connectors ensure the accuracy of mounting much higher. If the optical fiber is ended into an optical fiber terminator for redistribution this end connector is also called pig-tail and is prefabricated type. A special category of connectors is optical cords for distribution or connection. These are special optical fibers with connectors at both ends allowing small fiber curvature radii of approximately 2,5-5 cm. Their color is yellow for single-mode fiber and orange for multimode fiber.

Repeaters are optical amplifiers receiving light signals attenuated as a result of the distance traveled through optical fiber, remake the form, power and time parameters of these signals and send them away.

Patch panels for optical fiber are similar with copper cable patch panels, increasing flexibility of the optical networks. For connecting different equipment, an optical fiber patch cord is used (also known as a zip cord - two flexible optical fibers with connectors at each end).

Additionally, several other active or passive devices are used with optical fibers (e.g.: optical couplers - combines or splits optical signals; optical attenuators - reduce the power level of an optical signal; optical isolators; fiber-optic switches; optical multiplexers, etc.).

The ISO/IEC 11801-1 specifies the requirements for coaxial, twisted-pair copper and optical fiber. The ISO/IEC 11801 (Europe) and ANSI/TIA-568-C (USA and Canada) standards define 7 classes of optical fibers (single-mode and multimode) as shown in Table 2.1, together with several important parameters (optical fiber requirements, the cable transmission performance and the physical cable requirements):

Table 2.1 *Optical fiber characteristics*

		Multimode					Single-mode	
Type		<i>OM1</i> 62,5/125 μm	<i>OM2</i> 50/125 μm	<i>OM3</i> 50/125 μm	<i>OM4</i> 50/125 μm	<i>OM5</i> 50/125 μm	<i>OS1</i> 9/125 μm	<i>OS2</i> 9/125 μm
Wavelength		850, 1300nm	850, 1300nm	850, 1300nm	850, 1300nm	850, 1300nm	1300nm, 1550nm (1383nm)	1300nm, 1550nm
Max. attenuation (db/km)		2.6 / 2.4	3.56 / 2.3	2.6 / 1.9	2.9 / 1.5	2.9 / 1.5	1	0.4
Light source		LED (Light-Emitting Diode) / VCSEL (Vertical Cavity Surface-Emitting Lasers Light Source)					LASER (Light Amplification by Stimulated Emission of Radiation)	
Distance/ data rate	1 Gbps	275m	550m	-	-	-	5-120km	
	10Gbps	33m	82m	300m	400m	400m	10-80km	
	40-100 Gbps	-	-	100m	150m	150m	2-80km	
Color		orange/ slate	orange	aqua	violet/ aqua	green/ lime	yellow	yellow

Incorrect installation of optical fiber has as result the increase in attenuation for the optical signal (improper installation of optical fiber may cause cracks in the heart to disperse the signal light). Excessive stretching or bending of the optical fiber may cause small cracks of the core which will scatter the light signal. Excessive bending of the optical fiber may have as a result the drop in incident angle of the light signal under critical angle of total reflection. For the connector installation the heads must be cut off and finished. After installation, the heads of the optical fibers, the fiber connectors and ports must be kept clean so that no attenuation will be introduced. Before use of optical fiber cables, their attenuation must be tested. At the design of an optical-fiber links, loss of power signal that can be tolerated must be calculated. This is called the budget of loss of optical link. Loss of power is measured in decibels (dB).

For optical fiber link testing there are several methods: continuity testing, visual fault locator, measurement of optical power output, OTDR and BER test error rate.

Continuity testers are used to test the continuity in an optical fiber. A visual fault locator (VFL) tool allows a technician to identify breaks, macrobends (refers to the minimum bending radius) or poor fusion splices.

The measurement of optical power output determines the loss of power through the optical link by measuring the output power at a known input power. The unit of measurement for optical power is the milliwatt (mW) but for practical reasons shall be used other unit of measure which measure the gain (G) or loss (L) in a system, namely decibel (DB).

The procedure OTDR Optical Time Domain Reflectometer is the procedure by which the attenuation characteristics of an optical fiber and its length may be visualized. This procedure is the only through which can be detected positions such breaks in optical fiber. OTDR displays a graphic having as X axis the fiber length and as Y axis the attenuation. From this graphic, the fiber attenuation and the splices and connectors quality can be deduced. Also, can be determined the braking position in the cable if externally the cable is not affected.

The BER test (Bit Error Rate) is the final test for a data link through optical fiber. This test or criterion shows at how many bits transmitted through the fiber an error due fiber will be produced. The BER test must meet the requirements imposed by the producers of the DTE equipment that are coupled to the optical fiber. For computer networks they ask to be less than 1 bit of error at $10^9/10^{12}$ bits transmitted or $BER < 10^{-9}/10^{-12}$. For the testing is required a generator of random bit sequence and an interface to optical fiber if a loop is tested or two if a single fiber is tested. In order to have significant results, the test must be carried out over a period long enough so as to provide a sufficient number of bits. The test period of one day or two are common if it is working at a large bit rate in the use of optical fiber link and small BER. A counter may automatically count the number of errors detected.

Computation of **optical power budget** shall be made according to the Table 2.2.

Table 2.2 *Optical power budget*

Crt.	Optical loss or power	DB
1.	The km loss in Optical Fiber ___dB /km X _____ km fiber	_____dB
2.	The loss in Splices ___dB/splice X _____ splices	_____dB
3.	The loss in Connectors __dB/connector X ___ connectors	_____dB
4.	Losses on other components	_____dB
5.	Margin of error	_____dB
6.	Total loss on the Link (1+2+3+4+5)	_____dB
7.	The power of average emission of the transmitter	_____dB
8.	Average power received by the receiver (7-6)	_____dB
9.	The dynamic of the receiver _____dB at _____dB	
10.	Receiver sensitivity at a rate of errors given by BER	_____dB
11.	Available Remaining Power (8-10)	_____dB

Remarks

For item 3. the transmitter connection losses to the optical will not be taken into account, these being already included. The amount calculated in item 8. must be within the range of item 9. for the receiver to operate correctly. The amount calculated in item 11 must be positive in order to have a functional optical data link.

The error margin is due to take into account the average values for all link components. The dispersion of these values around the mean value is known and may take a margin of error large enough to cover deviations from an average with a probability of 99.9% or more. As the number of items is greater and as it is desirable a larger cover probability than a larger error margin will be taken.

Optical emission power of the transmitter is a catalogue data and includes the loss of connection at one end of the optical fiber in the case in which the connection is made in accordance with recommendations. The power is greater at the LASER diodes and smaller at the LED. In the case of LASER usage for relatively short distances an attenuator is necessary so that the receiver will not be destroyed.

Receiver dynamics represents the power range which a receiver can transform in electrical signal without loss of information.

It is also needed a minimum optical power necessary for fulfilling the tolerated error rate condition which for computer networks is situated at the value of 1 bit erroneous at one billion bits transmitted.

Calculus example of the optical power budget

Optical fiber diameter: Core 62.5 μ m/Cladding 125 μ m.

Numerical aperture of the fiber NA: 0.275.

The wavelength of the optical equipment: 1310 μ m.

The solution is presented in Table 2.3.

Table 2.3 *Calculus example*

Crt.	Optical loss or power	DB
1.	The km loss in Optical Fiber 1,8 dB/km X 3,5km fiber	6,3dB
2.	The loss in Splices 0,5 dB/splice X 2 splices	1,0dB
3.	The loss in Connectors 1,0 dB/connector X 2 connectors	2,0dB
4.	Losses on other components	0,0dB
5.	Margin of error	2,0dB
6.	Total loss on the Link (1+2+3+4+5)	11,3dB
7.	The power of average emission of the transmitter	-10,0dB
8.	Average power received by the receiver (7-6)	-21,3dB
9.	The dynamic of the receiver _____ dB at _____ dB	
10.	Receiver sensitivity at a rate of errors given by BER	-26,0dB
11.	Available Remaining Power (8-10)	+4,7dB

The power at the receiver is in the dynamic of the receiver, which makes possible its function, and the remaining available power is positive, ensuring a viable connection.

There should be taken into account the fact that during the life of the link, aging phenomena may occur, leading to increase the power loss, as well as the fact that optical fiber may be broken accidentally and needs to be spliced.

A calculation made to the limit endangers the length of service of a link through optical fiber.

3. Practical activity

3.1 The characteristics of various types of optical fibers, components and aspects related to the cabling of computer networks using this transmission environment should be discussed.

3.2 Explore the fiber optic infrastructure deployed in the oceans available at <https://www.submarinecablemap.com/>

3.3 A 9/125 μ single-mode optical fiber having the length of 2,5km and the loss equal to 0,5dB/km, which connects two DTE equipments is considered. The attenuation introduced by splices and connectors is equal to 0,5 and 1dB respectively. The error margin taken into consideration is 3dB. The power of average emission of the transmitter is -15dB, the receiver sensitivity at a rate of errors given by BER 10^{-9} is -25dB and dynamic of the receiver is in the range -10 ÷ -30dB. Calculate the optical power budget.

CHAPTER 4: STRUCTURED CABLING

1. Objectives

The objective of this paper is the knowledge of structured cabling, networks topology and the function of the different network devices.

2. Theoretical considerations

The current practical work focuses on the Physical, Data Link and Network layers of the ISO/OSI stack (Figure 4.1) by presenting the main elements of structured cabling and network devices.

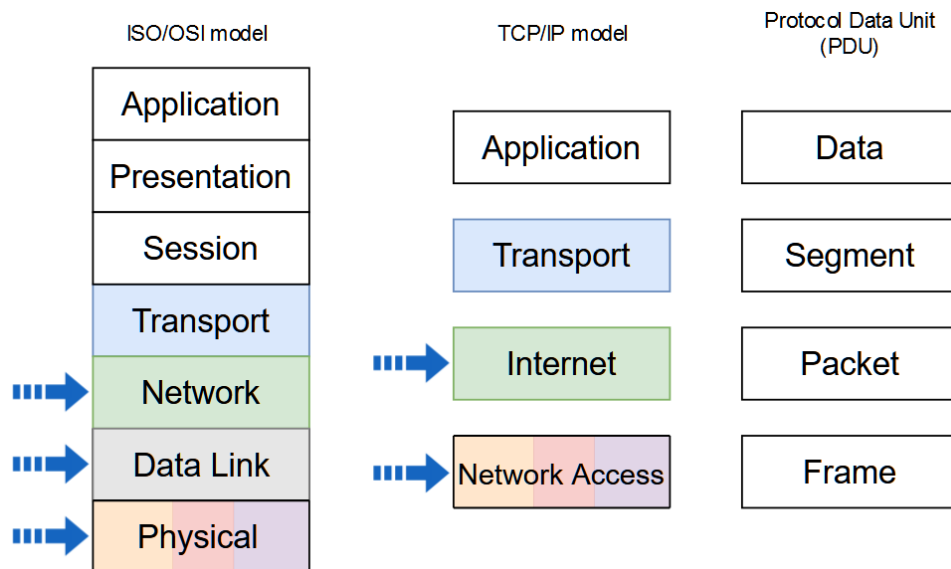


Figure 4.1 Network stack models and PDU naming in each level. The arrows indicate the addressed layers in the current activity

2.1 Physical media analysis

In the physical media analysis, we may choose several factors of performance such as: the speed of transfer, bandwidth, reliability or the error rate, the duration of service, the average duration between the two defects, defects tolerance, direct costs, indirect costs, the cost per port or equipment connected, the cost per bandwidth or the total cost per port per bandwidth. The bandwidth, L_B is a factor of intrinsic performance particular to each medium. The reliability, F , is also a factor of intrinsic performance of each medium and shall be the ratio of the number of bits erroneously transmitted to the total number of bits transmitted. The service duration, D_e , is the length of time after the environment should be replaced, due to aging phenomena. The average duration between two faults, $DMDD$, is the statistical average time

between two successive malfunctions of the environment for the standardized period of life. Defects tolerance, T_d , is a factor of performance induced on the physical environment by the technology and network architecture used, but in many cases, a given environment does not allow a error tolerance architecture or only one limited. Direct costs, C_d , are represented by the actual cost of the environment along with connectors, the auxiliary materials necessary for correct posing, and the cost of labor for communication environment realization and environment testing. The cost per port, C_p , it is a synthetic factor which has a greater decision value, being a global decision criterium and reflecting the total costs for carrying out physical infrastructure related to the total number of ports or equipment connected. The cost per port per speed of transfer, C_{pv} , is a factor performance more useful which alleviates taking a correct decision in the implementation of a local area network, including the possibility of future extension without the need for change the environment. The total cost per port per speed, C_{tpv} , is a complex factor of performance which characterizes a local area network at global level also including the equipment or technology costs. Characterization of performance factors above referred of the physical communication media previously presented is summarized in Table 4.1. Performance factors, and in particular the type of cost, shall be classified relatively without giving absolute values which may be affected very rapidly in time.

Table 4.1 Performance factors

Medium	L_b Gbps	Reliability	D_e years	DMDD	T_d	C_d	C_p	C_{pv}	C_{tpv}	Recommended in usage	Further use
UTP Cat 6,7	>1	Medium	15	years	Yes	Medium	Small	Small	Small	Yes	Yes
Multimode OF	>1	Large	30	years	Yes	Large	Medium	Medium	Medium	Yes	Yes
Single- mode OF	>1	Large	30	years	Yes	V. Large	V. Large	Large	Large	Yes	Yes

2.2 Structured cabling

There are three standard network topologies bus, star and ring (Figure 4.2):

- **Bus topology** is the oldest method of interconnecting computers in a network. Data is transmitted to all the stations but is accepted only by the destination station, and the reflection of the signal is stopped using terminators. Figure 4.2 a. represents the bus topology;
- **Star topology** has replaced the bus topology, the main feature is that it has a central component called hub through this component data is transmitted from one station to all the others. The star topology offers the resources and means for central administration. Figure 4.2 b. represents the star topology;
- **Ring topology** stations are connected through a cable shaped as a ring and every station is acting as a repeater amplifying the signal. Figure 4.2 c. represents the star topology.

Today most of the topologies used are combinations of star, ring and bus topologies. The bus-star topology supposes connecting networks with star topology through linear branches (bus). Problems of connectivity appear when a concentrator fails. The ring-star topology also known

as ring cabled as a star. In this case there is a central concentrator that connects all the other concentrators to which the stations are connected.

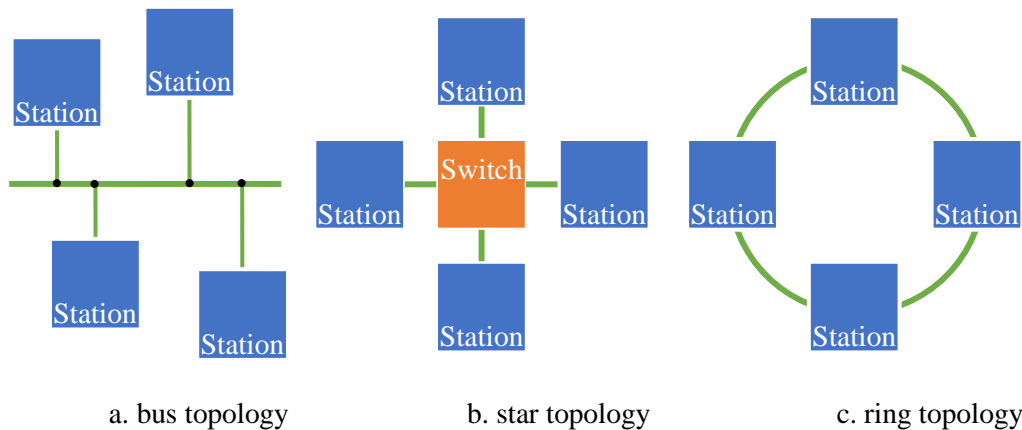


Figure 4.2 *Standard network topologies*

Under the generic name of active elements are grouped all of the network components that need a power supply and can work with electric, optic signals or both. **Network interface cards** are active elements of layer 2 providing the stations with the network connectivity. Every network interface card has its own 48 bits MAC address assigned from fabrication. This address is unique for every network card and it is composed of 2 parts: the 24 most significant bits identifies the producer, and the 24 least significant bits are assigned by the producer. The network interface cards used in PC's need an I/O address space and a hardware interrupt. The interrupt is activated every time an event (a frame reception in most of the cases) appears requiring software attention, and the I/O address space represents the address region in which the card registers are accessible (written, read, by its driver). Usually both the interrupt and the I/O space are configurable to avoid conflicts with other devices.

The overcome of the length limitations of cables is done by using **repeaters**. These are simple devices, connected at many network segments amplifying the signal that passes through them. Repeaters operate at the physical layer (they don't have the frame notion or package transmitted through the network) and they broadcast the amplified signals on all their outputs.

With the growth of the network dimensions, problems will appear if there are used only repeaters. The limitation for the stations that create such a network is the fact that repeaters/hubs (multiport repeaters) split the bandwidth, being situated in a single **collision domain**. In order to solve this problem, we use a **bridge**, equipment that operate at the second layer in the OSI hierarchy, and they represent devices much more complex than repeaters because they perform frame filtering based on MAC addresses and a separation of collision domains. Bridges don't forward the frames that are local for a network, but only the ones that have destination addresses located in other networks. They store the frames and realize a retransmission only to the network in which the destination is situated. When the bridges are powered-up they know nothing about the network configuration and the addresses of the computers connected to it, but they learn the network topology while they forward the frames. Initially they allow all the frames to pass in all directions. But in time, as frames pass through, the bridge inspects the source address of each frame and completes the MAC tables, with the station address and the port at which the station is connected. Based on these tables they decide on which port the frames must be retransmitted. Frames sent at broadcast or multicast addresses

will be retransmitted further away on all ports. A **switch** is a layer 2 equipment that take frames forwarding decisions based on the MAC address, so to direct the data only on the port corresponding to the destination host. These devices can be seen as devices capable to offer the connectivity of a hub and they manage the traffic like a bridge. Designing networks with complex topologies is done using switches.

A **router** is a layer 3 equipment that route the packets based on the address used by routable protocols (for example Internet Protocol-IP or Internetwork Packet Exchange – IPX) with the help of the routing protocols (for example Routing Information Protocol RIP, Interior Gateway Routing Protocol – IGRP, Enhanced Interior Gateway Routing Protocol – EIGRP or Open Shortest Path First - OSPF). There are two main router types: dedicated routers and routers built from general purpose computers that have more interfaces. The computer routers have the advantage of cost and simplicity and can be used for other jobs. Dedicated routers are much more efficient and flexible, have much more interfaces and support more protocols and medium access types. Dedicated routers are devices specialized for the routing job. Due to the specialized hardware and powerfully optimized software, they achieve superior performance. They offer a wide range of speeds, physical interfaces and communication protocols. Usually these are manufactured by specialized firms (Cisco, Juniper, HPE etc.) their operating system is specific and has all the software need for the router to function properly. Dedicated routers support almost any transmission medium, used with any communication protocol, with a large range of sockets and adaptors.

Taking in consideration the costs for realizing or modifying a network cabling it has been proved that once a network has been set in place is better to stay in use as long as possible and that it should be able to be used with novel communication technologies. The solution for this problem was in the elaboration of the **structured cabling** concept, defined later through several international standards.

The ISO/IEC 11801 (Europe) and ANSI/TIA-568-C (USA and Canada) standard refer to the ways of cabling commercial edifices, specifying the cabling structure, the necessary minimal configuration, the categories of cables and components that must be used, ways of installation, performance requests that have to be met, acceptable distance limits and other parameters, and also ways and methods for testing them. Another problem that is approached is the problem of designing the cabling for a much more complex building group, in this way a complex project needs to be configured in a hierarchic (tree-like) structure, allowing the possibility to add redundant links. The standard specifications refer to some of the following aspects:

- Minimal request for realizing the cabling of a building
 - The cabling topology and allowed distances;
 - Component elements of the cabling;
 - Transmission media used with the needed parameters specification;
 - Vertical and horizontal cabling realization mode;
 - Ways of identifying the cables used;
 - Project documentation.
- Subsystems and components of the structured cabling system
 - The subsystem from the entrance in the building;
 - The equipment room;
 - The backbone cabling;
 - The telecommunication closet;
 - Horizontal cabling;
 - The work area components.

The cabling topology specified in the ISO/IEC or ANSI/TIA standard is a star, hierarchically organized (extended star). The topology center is main distribution facility, the second hierarchic level is the intermediary distribution facility afferent to one area edifice, and at the lower level is the telecommunication closet related to a floor or a group of rooms. The constitutive elements are:

- *The main distribution facility* – the distribution center to the other edifices;
- *The intermediary distribution facility* – are local to edifices;
- *The telecommunication closet* – is represented by the local distribution closets for the cables that connect the stations or related to the vertical cabling;
- *The inter-edifice section* – identifies the main cables that interconnect the main distribution center;
- *The internal section* – connects the intermediate commuter with the distribution offices;
- *The equipment room* – related to a cabling plan with passive and active devices;
- *The entrance infrastructure* – for the interfacing of the exterior cabling system with the interior one;
- *The work area* – the working stations, interconnection cables, external adaptors between cables;
- *Intermediate panels* – identifies the connection panels for the transmission mediums;
- *Terminator blocks* – represent the cable mechanical terminators;
- *Communication outlets, cabling adaptors.*

The usual transmission media are:

- Twisted cable (category 6 and above);
- Multimode or single-mode optical fiber;

Types of the connectors used are:

- RJ-45 connectors for TP cables;
- LC, SC or ST type connectors for optical fiber;

So, in order to accommodate a much easier and efficient way to manage the network, the cabling is structured using concentrators (on different levels). At each level a concentrator must be implemented, and if the covered area is too large than several concentrators can be used. At the working stations the UTP cable is ended in RJ-45 connectors, and at the concentrator in boxes or patch panels. The cumulative length of the cable and UTP patch cord used for connecting a computer at the equipment from the concentrator is not allowed to be greater than 100m. In the floor concentrator the switches or other devices are situated.

The advantages concentrators offer (and also the topologies based on concentrators) are:

- possibility to extend or modify the cable system;
- usage of different ports, adapted at different types of cables;
- possibility of a central monitoring of the activity and the network traffic.

Types of concentrators:

- **Active concentrators** – that regenerates and transmits the signal;
- **Passive concentrators** – can be considered the cabling panels or the connection blocks representing only connection points without any signal amplification. Also

there are hybrid concentrators that allow the usage for connection of different cable types.

The cables must be labeled according to the standard, the ventilation must be sufficient to prevent equipment overheating, security measures must be set and fire protection must be provided. The floor concentrator is connected to the building concentrators, link that can be realized with a category 6 cable or with multi-mode optical fiber. Additionally, redundant links can be added between the floor concentrators and between the buildings. The building group concentrator is connected to the buildings concentrators with multimode or single-mode optical fiber. Installation standards are referring to the cable installation (maximum tension allowed on the cable, mechanical connection type), masked horizontal cabling, ground protection, and the specific protection of the optical fibers cables.

3. Practical activity

3.1 The topologies of the computer networks are going to be discussed underlining their advantages and disadvantages.

3.2 The function of the following network devices will be discussed: network interface card, concentrator, repeater, bridge, switch and router.

3.3 Aspects of the structured cabling and ISO-IEC/ANSI-TIA standard will be discussed.

3.4 Floor cabling will be analyzed, and the elements of the structured cabling will be pointed out.

3.5 Identify and analyze the structured cabling design at your workplace/home. How is your network connected to the WAN/ISP (what type of cable, device, etc.)? How is your device connected to the internal network?

CHAPTER 5: NETWORK LAYER – IPv4 FUNDAMENTALS

1. Objectives

At the end of the practical activity, students will be able: to explain the characteristics of the network layer, to describe the operation of the IPv4 protocol, to divide the networks into subnets, to explain the network address translation process, and to implement basic IPv4 network configurations.

2. Theoretical considerations

The current practical work focuses on the Network layer of the ISO/OSI stack (Figure 5.1).

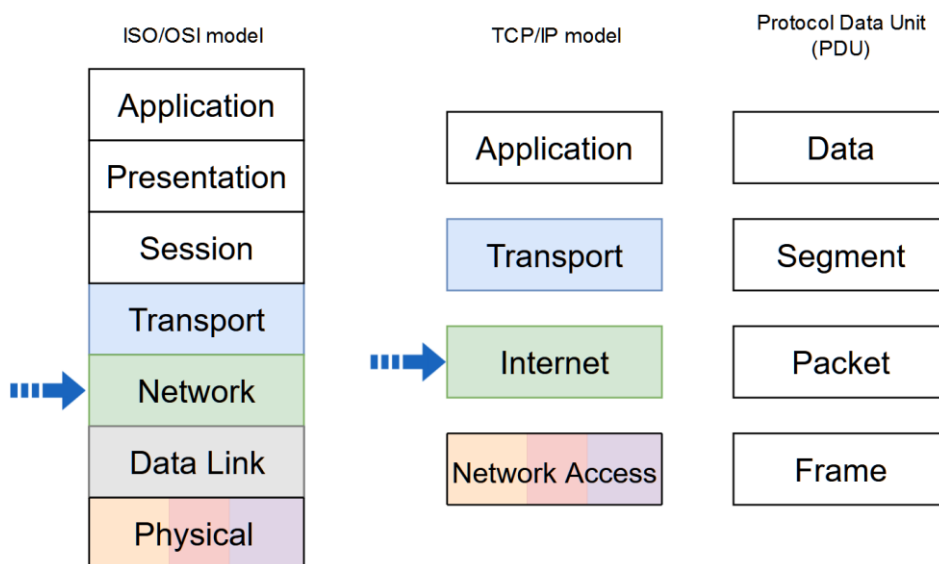


Figure 5.1 Network stack models and PDU naming in each level. The arrows indicate the addressed layers in the current activity

2.1 Network layer

The OSI Network layer corresponds to the TCP/IP Internet layer. It provides addressing, routing and traffic control services to allow devices to exchange data across networks and contains different types of protocols:

- IP version 4 (IPv4) and IP version 6 (IPv6) routed protocols;
- routing protocols such as Open Shortest Path First (OSPF) or Border Gateway Protocol (BGP);
- messaging protocols such as Internet Control Message Protocol (ICMP).
- The network layer performs four basic operations (Figure 5.2):
- Addressing
- Encapsulation
- Routing
- De-encapsulation

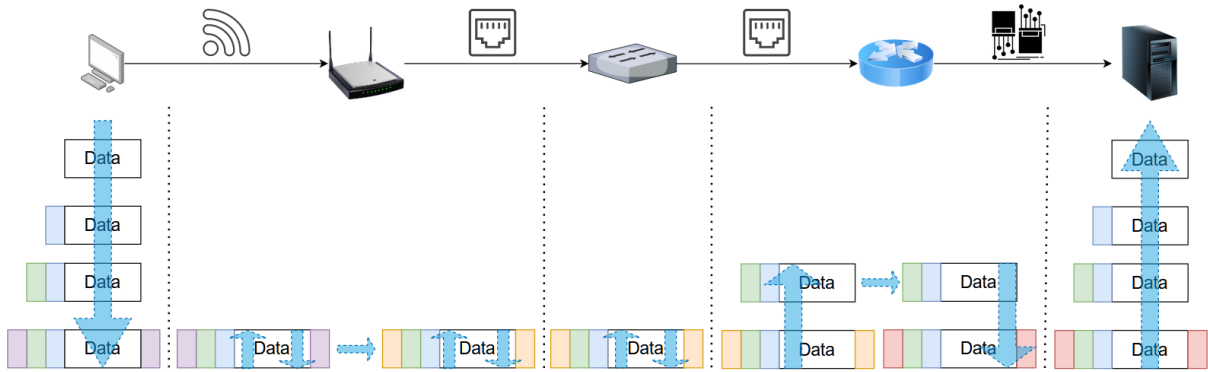


Figure 5.2 Network layer operations and packet serialization/deserialization when passing through different network devices

IP protocols have the following characteristics:

- Connectionless
 - no connection established between source and destination before data packets transmission;
 - no control information (synchronizations, acknowledgments, etc.).
- Best Effort
 - unreliable, packet delivery is not guaranteed;
 - no mechanism to resend data that is not received, reduced overhead.
- Media Independent
 - does not concern itself with the type of frame required at the data link layer or the media type at the physical layer;
 - can be sent over any media type: copper, fiber, or wireless.

2.2 IPv4

The packet header is presented in Figure 5.3:

Octet	0							1					2				3															
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	Version				IHL			DSCP				ECN	Total Length																			
32	Identification										Flags		Fragment Offset																			
64	Time To Live					Protocol					Header Checksum																					
96	Source IP Address																															
128	Destination IP Address																															
160	Options																															

Figure 5.3 IPv4 packet header

- Version - version field, equal to 4;
 - Internet Header Length (IHL) - the size of the IPv4 header;
 - Differentiated Services Code Point (DSCP) - originally defined as the type of service (ToS), specifies differentiated services (DiffServ);
 - Explicit Congestion Notification (ECN) - allows end-to-end notification of network congestion without dropping packets, optional feature;
 - Total Length - defines the entire packet size in bytes, including header and data;
 - Identification- identification field, primarily used for uniquely identifying the group of fragments of a single IP datagram;
 - Flags - used to control or identify fragments;
 - bit 0 – Reserved, must be zero;
 - bit 1 – Don't Fragment (DF)
 - bit 2 – More Fragments (MF)
 - Fragment offset –specifies the offset of a fragment relative to the beginning of the original unfragmented IP datagram;
 - Time to live (TTL) – limits a datagram's lifetime;
 - in practice, is used as a hop count;
 - when the datagram arrives at a router, the router decrements the TTL field by one;
 - when the TTL field hits zero, the router discards the packet and sends an ICMP time exceeded message to the sender.
-
- Protocol – defines the protocol used in the data portion of the IP datagram;
 - Header checksum – used for error-checking of the header;
 - Source address – the IPv4 address of the sender of the packet;
 - Destination address – the IPv4 address of the receiver of the packet;
 - Options – rarely used; if IHL is greater than 5, the options field is present.

IPv4 addresses can be assigned statically or dynamically.

The IPv4 address is hierarchical, being composed of two parts: the network part and host part (Figure 5.4).



Figure 5.4 IPv4 address structure

The number of bits assigned to the network and host depends on the class / network mask to which the address belongs (Table 5.1):

Table 5.1 IPv4 network classes

Class	1 st Octet Decimal Range	1 st Octet High Order Bits	Network/Host ID (N=Network, H=Host)	Default Subnet Mask
A	1 – 126*	0	N.H.H.H	255.0.0.0
B	128 – 191	10	N.N.H.H	255.255.0.0
C	192 – 223	110	N.N.N.H	255.255.255.0
D	224 – 239	1110	Reserved for Multicasting	
E	240 – 255**	1111	Experimental; used for research	

Note: * Class A addresses 127.0.0.0 to 127.255.255.255 cannot be used and is reserved for loopback and diagnostic functions.

** 255.255.255.255 is reserved as the IPv4 Broadcast address.

The IPv4 subnet mask (Figure 5.5) is used to differentiate the network portion from the host portion of an IPv4 address. It is, like the IPv4 address, a 32 bits structure. The bits corresponding to the network portion are set to 1 and the bits corresponding to the host portion are set to 0.



Figure 5.5 IPv4 subnet mask

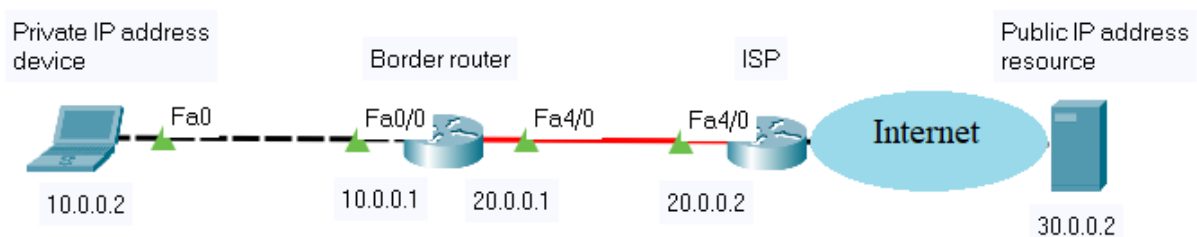
The network masks corresponding to the classes are presented below:

- Class A: 255.0.0.0 or /8 (11111111.00000000.00000000.00000000)
- Class B: 255.255.0.0 or /16 (11111111.11111111.00000000.00000000)
- Class C: 255.255.255.0 or /24 (11111111.11111111.11111111.00000000)

Public IPv4 addresses are uniquely assigned addresses and are globally routed between internet service provider (ISP) routers. There are also blocks of addresses, called private addresses, that are used by most organizations to assign IPv4 addresses to internal hosts. These addresses are not uniquely assigned addresses and are not globally routed between ISP routers. These blocks of private addresses are presented below.

- Class A: 10.0.0.0 - 10.255.255.255 /8
- Class B: 172.16.0.0 - 172.31.255.255 /12
- Class C: 192.168.0.0 - 192.168.255.255 /16

To allow a device with a private IPv4 address to access devices and resources outside of the local network, the private address must be translated to a public address. This process is called network address translation (NAT) (Figure 5.6) and provides the translation of private addresses to public addresses. A NAT router typically operates at the border of a network. When a device inside the network wants to communicate with a device outside of its network, the packet is forwarded to the border router which performs the NAT process, translating the internal private address of the device to a public, outside, routable address.




```

BorderRouter#show ip nat translations
Pro  Inside global      Inside local      Outside local     Outside global
tcp  20.0.0.3:1027        10.0.0.2:1027    30.0.0.2:80      30.0.0.2:80

```

Figure 5.6 Network address translation

The network address has all the host bits set to 0 and the broadcast address has all the bits set to 1. These addresses cannot be assigned to a host. All the other addresses are valid host addresses.

Exercise

Consider the following address: 192.168.1.10/24. Calculate the network and broadcast address, the valid host range, the total number of host bits and the total number of hosts.

IP: 11000000.10101000.00000001.00001010

NM: 11111111.11111111.11111111.00000000

----- **logic AND IP with the NM:**

11000000.10101000.00000001.00000000 – Network address (all host bits are set to 0)

192.168.1.0 – **Network address** (decimal notation)

11000000.10101000.00000001.11111111 – Broadcast address (all host bits are set to 1)

192.168.1.255 – **Broadcast address** (decimal notation)

11000000.10101000.00000001.00000001 – First valid host address

192.168.1.1 – First valid host address (decimal notation)

11000000.10101000.00000001.11111110 – Last valid host address

192.168.1.254 – Last valid host address (decimal notation)

Results:

192.168.1.1-192.168.1.254 – **Valid host range** (decimal notation)

Total number of host bits is 8.

Total number of hosts is $2^8-2=254$.

2.3 Subnetting

To create subnets, bits are borrowed from the host ID. A new network mask (Figure 5.7) is created to show the new structure. In the network mask, the bits corresponding to the subnetwork portion are set to 1.

Network ID	Host ID	
11.....1	00.....0	
Network ID	Subnetwork ID	Host ID
11.....1	11.....1	00.....0

Figure 5.7 IPv4 subnet mask

Exercise

Consider the following address: 192.168.1.0/24. Divide this address in 4 subnets and further divide the fourth subnet into a maximum number of subnets. Specify for the subnets: netmask, network address, broadcast address, the number of host bits, the number of hosts and their address range.

2 bits are borrowed to obtain 4 subnets. In order to further divide the fourth subnet into a maximum number of subnets, another 2 bits will be reserved for the host portion, the minimum possible number (Figure 5.8).

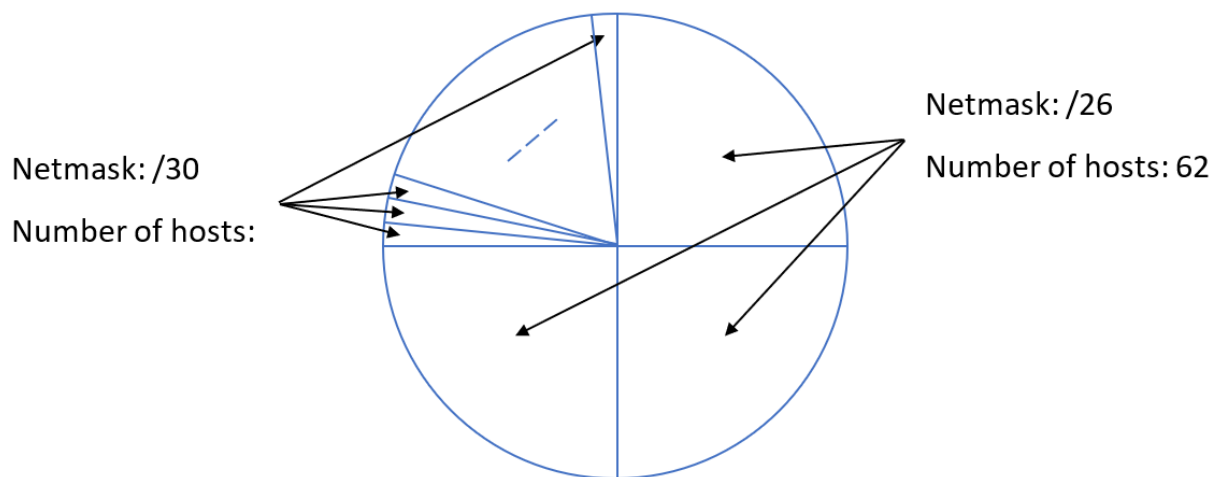


Figure 5.8 Dividing a network address into subnets

First /26 subnet:

Network Subnetwork Host	
11000000.10101000.00000001.00000000	192.168.1.0/26 - Network Address
11000000.10101000.00000001.00000001	192.168.1.1/26 - First Host Address
...	...
11000000.10101000.00000001.00111110	192.168.1.62/26 - Last Host Address
11000000.10101000.00000001.00111111	192.168.1.63/26 - Broadcast Address

Netmask: /26 (255.255.255.192)

Network address: 192.168.1.0/26

Broadcast address: 192.168.1.63/26

Number of host bits: 6

Number of hosts: $2^6-2=62$

Hosts address range: 192.168.1.1/26-192.168.1.62/26

First /30 subnet:

Network Subnetwork Host	
11000000.10101000.00000001.11000000	192.168.1.192/30 - Network Address
11000000.10101000.00000001.11000001	192.168.1.193/30 - First Host Address
...	...
11000000.10101000.00000001.11000010	192.168.1.194/30 - Last Host Address
11000000.10101000.00000001.11000011	192.168.1.195/30 - Broadcast Address

Netmask: /30 (255.255.255.252)

Network address: 192.168.1.192/30

Broadcast address: 192.168.1.195/30

Number of host bits: 2

Number of hosts: $2^2-2=2$

Hosts address range: 192.168.1.193/30-192.168.1.194/30

3. Practical activity

3.1 Discuss the theoretical aspects of the chapter. What is the difference between classful and classless addressing?

3.2 Solve the following problems:

A. Determine the network and broadcast addresses and number of host bits and hosts for the given IPv4 addresses and prefixes (Table 5.2):

Table 5.2 IPv4 addresses and prefixes

IPv4 Address/Prefix	Network Address	Broadcast Address	Total Number of Host Bits	Total Number of Hosts
172.16.104.99/27				
198.133.219.250/24				
10.1.113.75/19				

- B. Having the following information, compute subnets with the following constraints:
- A number of 62 subnets
 - Host IP Address: 172.16.0.0
 - Original Subnet Mask: 255.255.0.0
- C. Having the following information, compute subnets with the following constraints:
- A maximum number of 29 hosts/subnet
 - Host IP Address: 192.168.200.0
 - Original Subnet Mask: 255.255.255.0
- D. Having the following information, compute subnets with the following constraints:
- A number of 250 subnets
 - Host IP Address: 10.0.0.0
 - Original Subnet Mask: 255.0.0.0

3.3 Test the following commands (using Command Prompt on Windows OS or Terminal in Linux OS):

- Command: **ipconfig /all** (on Windows OS) and **ifconfig** (on Linux OS)
 - Role: displays all network configuration values for your network interface cards
- Command: **ipconfig /release** and **ipconfig /renew** (on Windows OS) and **dhclient** (on Linux OS)
 - Role: refreshes DHCP and DNS values
- Command: **ping**
 - Role: troubleshoots network connectivity; verifies IP connections, using ICMP packets
- Command: **tracert** (**traceroute** on Linux)
 - Role: troubleshoots network connectivity; resolves the path to an IP destination, using ICMP packets
- Command: **nslookup**
 - Role: performs DNS queries
- Command: **route print**
 - Role: displays the routing table of the host device
- Command: **netstat**
 - Role: network statistics tool
- Command: **arp -a**
 - Role: displays the ARP cache (mapping of IP address to a physical addresses)

Hint: you can use online operating systems to test various commands (e.g. <https://bellard.org/jslinux/> for Alpine Linux or Windows 2000)

3.4 Using Wireshark, capture different types of IP packets and analyze their headers. For example:

- capture **ping** traffic by filtering the ICMP protocol filter
- capture **nslookup** traffic by filtering the DNS protocol filter
- etc.

3.5 Configure and test the following network (Figure 5.9) using Cisco Packet Tracer:

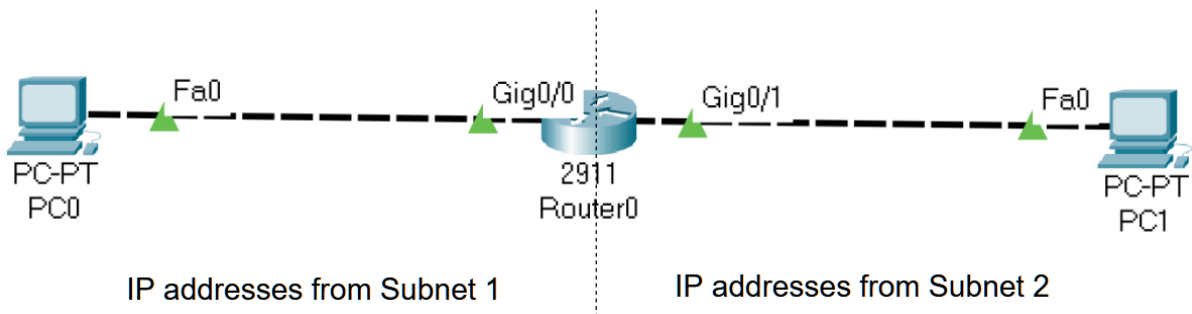


Figure 5.9 Test network topology

Considering IP address 172.16.0.0 /16, compute 2 subnets and assign the correct IP address to the router's interfaces and to the host computers (PC0 and PC1).

Step 0: In order to show the interface name and numbers, go to Options -> Preferences and check "Always Show Port Labels in Logical Workspace" (Figure 5.10)

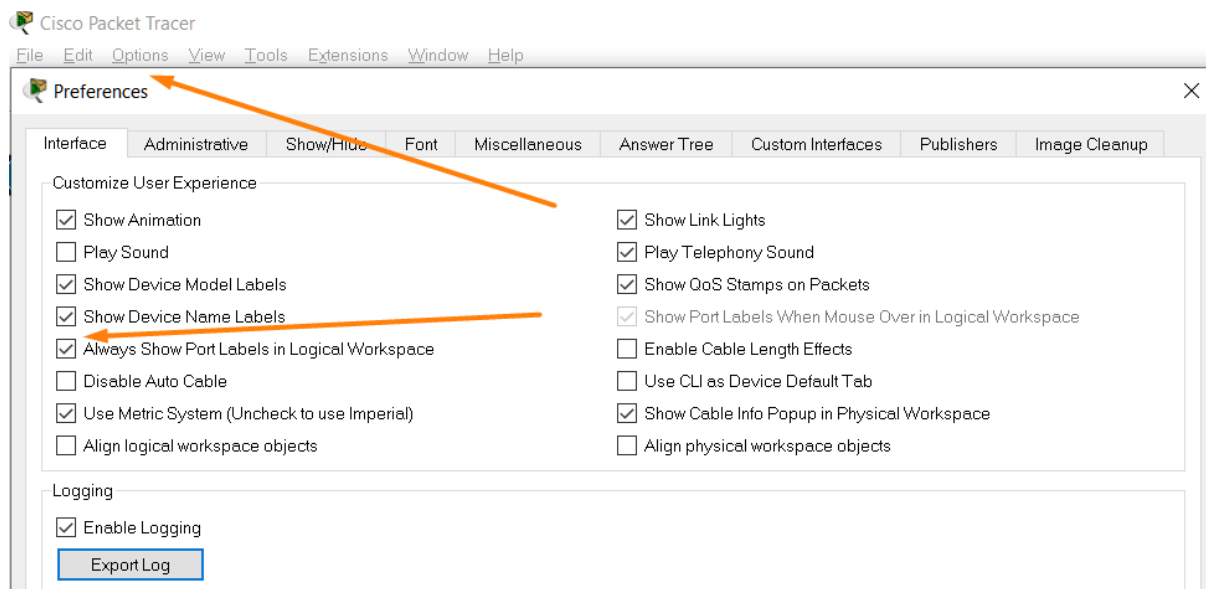


Figure 5.10 Preferences window

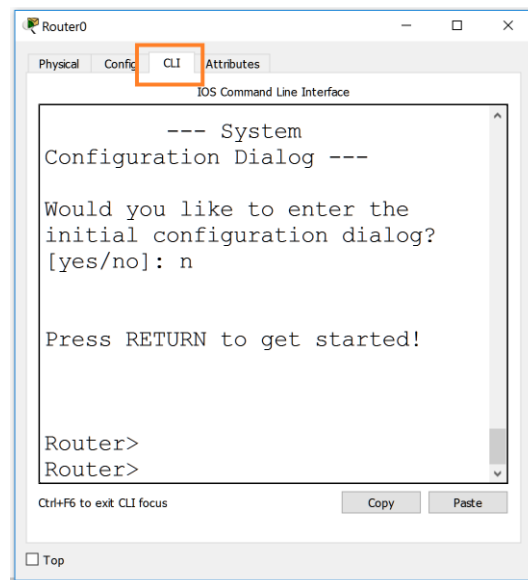
Step 1: Compute the two subnets needed.

Step 2: Before configuring the network devices, assign a unique IP address and the corresponding subnet mask to each network interface and fill in the Table 5.3:

Table 5.3 IPv4 addresses and netmasks for the test network

Device	Interface	IP Address	Subnet mask	Default Gateway
PC0	Fa0	172 . 16 . ____ . ____	255 . 255 . ____ . ____	172 . 16 . ____ . ____
Router0	Gig0/0	____ . ____ . ____ . ____	____ . ____ . ____ . ____	-
Router0	Gig0/1	____ . ____ . ____ . ____	____ . ____ . ____ . ____	-
PC1	Fa0	____ . ____ . ____ . ____	____ . ____ . ____ . ____	172 . 16 . ____ . ____

Step 3: Configure the router using the commands provided in the steps below. The commands provide sample interface names and IP addresses. You must use the interface names and the IP addresses filled in the Table 5.3. Figure 5.11 shows how the CLI of a router can be accessed.

**Figure 5.11** Accessing the router's CLI

Step 3.1: Enter configuration mode on the router

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

Step 3.2: Assign static IPv4 address to the router interfaces

```
Router(config)#interface GigabitEthernet 0/0
Router(config-if)#ip address 172.16.0.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#exit
```

Configure the other router interface with the corresponding IP address

```
Router(config)# interface ____
Router(config-if)#ip address _____
Router(config-if)#no shutdown
```

Step 3.3: Display information about the router configuration

```
Router#show ip interface brief
```

Description: Display IP information about router's interfaces

```
Router#show ip route
```

Description: Display IP routing table

Step 4: Configure IP addresses on the PCs using the screenshots in the Figures 5.12, 5.13 and 5.14.

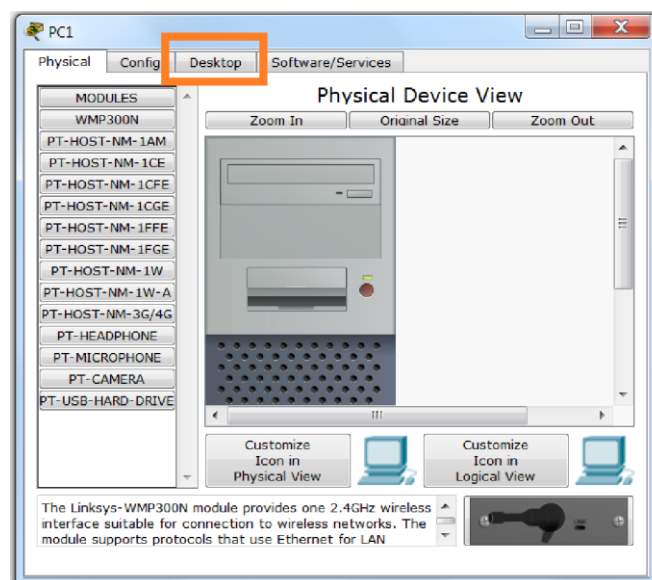


Figure 5.12 PC configuration screenshot



Figure 5.13 PC Desktop screenshot

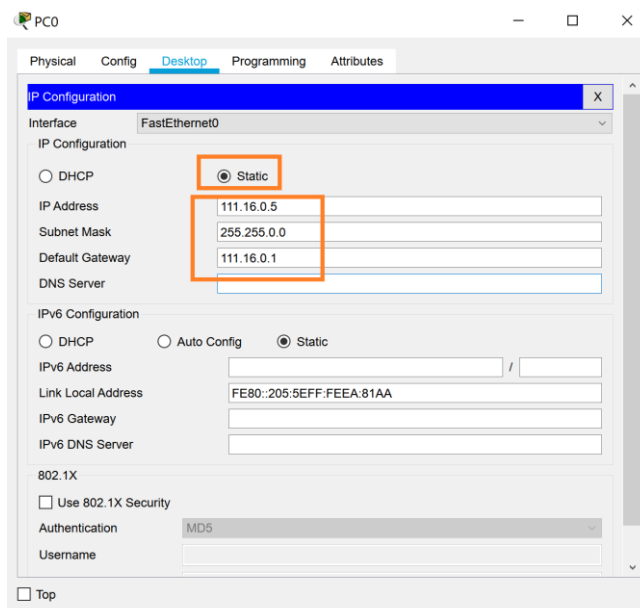


Figure 5.14 PC IP configuration screenshot

Step 5: Test the connectivity and troubleshoot if it is not working.

- a. check IP addresses oh hosts computers: PC -> Desktop -> IP Configuration
- b. Check connectivity between computers using the **ping <target IP>** command: PC -> Desktop -> Command prompt

CHAPTER 6: NETWORK LAYER – IPv4 ROUTING AND DHCP

1. Objectives

At the end of the practical activity, students will be able: to explain the routing process, to describe the operation of the DHCPv4 protocol, and to implement basic IPv4 network configurations.

2. Theoretical considerations

The current practical work focuses on the Network layer of the ISO/OSI stack (Figure 6.1).

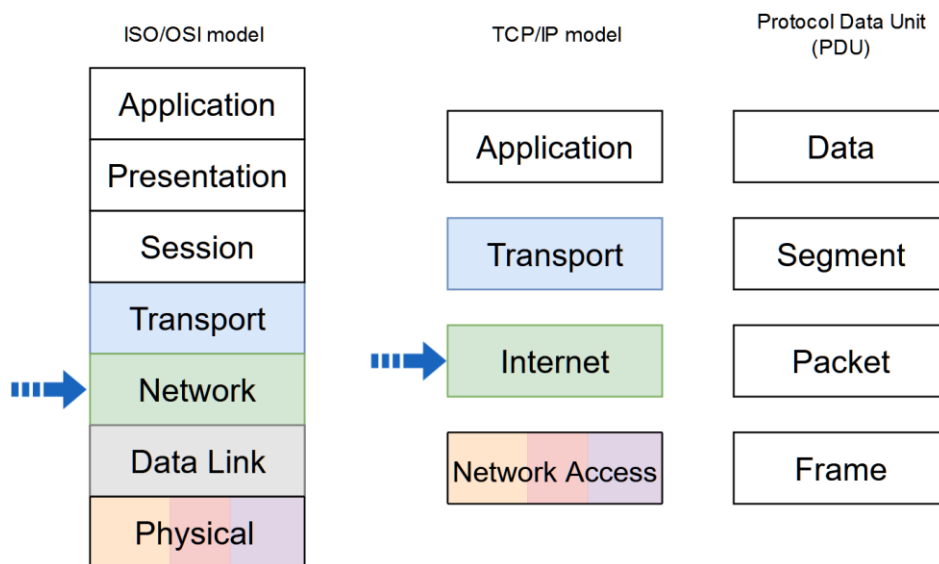


Figure 6.1 Network stack models and PDU naming in each level. The arrows indicate the addressed layers in the current activity

2.1 Routing

IP packets are created at the source host and are directed to the destination. Forwarding IP packets to the destination is based on the routing process. This is a distributed process: each node, which is forwarding packets based on the IP address, will choose the next node according to its own routing table. Forwarding IP packets is a hop-by-hop process, each node forwarding the packet to the next node.

Hosts and routers forward packets based on the destination IP address. Switches and access points are Layer 2 devices and do not forward packets based on the destination IP address (Figure 6.2).

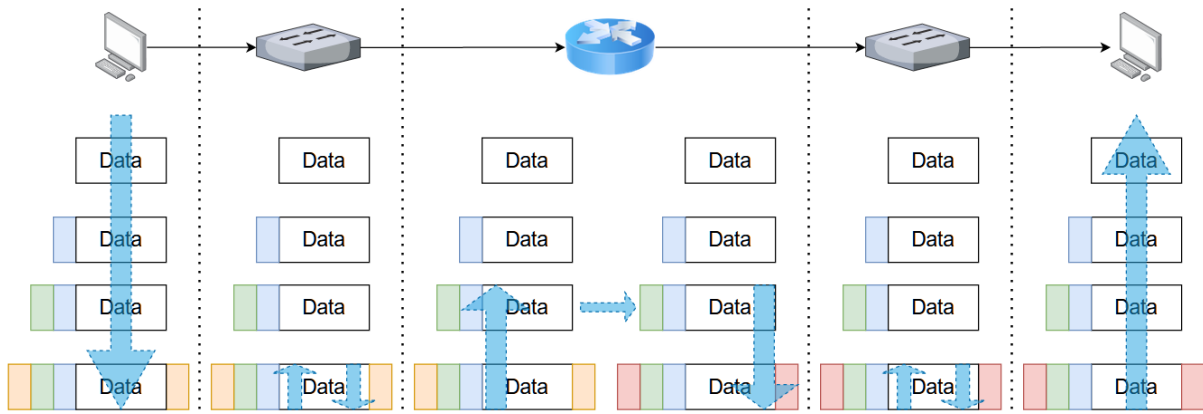


Figure 6.2 Network packet forwarding process showing packet serialization/deserialization when passing through different network devices

The host or the router examines the destination IP address of the packet and searches its routing table to determine where to forward the packet. The routing table contains a list of all known network addresses (prefixes) and where to forward the packets belonging to the corresponding networks. These entries are known as route entries or routes. The host or the router will forward the packet using the best (longest) matching route entry. The hosts and most routers also include a default route entry, 0.0.0.0/0. The default route is used when there is no better (longer) match in the IP routing table.

To identify the network address of an IPv4 host, the IPv4 address is logically ANDed, bit by bit, with the subnet mask. ANDing between the address and the subnet mask yields the network address. Exercise: find the network address for the host configured with 192.168.50.106 IPv4 address and 255.255.255.0 subnet mask.

IPv4 host address	192	.	168	.	50	.	106
	11000000		10101000		00110010		01101010
AND							
Subnet Mask	255	.	255	.	255	.	0
	11111111		11111111		11111111		00000000
IPv4 network address	192	.	168	.	50	.	0
	Equals		11000000		10101000		00110010 00000000

A host can send a packet to (Figure 6.3):

- Itself – to the loopback interface, 127.0.0.1 IPv4 address or ::1 IPv6 address;
- Local host - the destination host is on the same local network as the sending host, the source and destination hosts share the same network address;
- Remote host - the destination host is on a remote network, the source and destination hosts do not share the same network address.

The default gateway is the network device that can route traffic to other networks. It has a local IP address in the same address range as other hosts on the local network, accepts data into the local network, forwards data out of the local network and routes traffic to other networks.

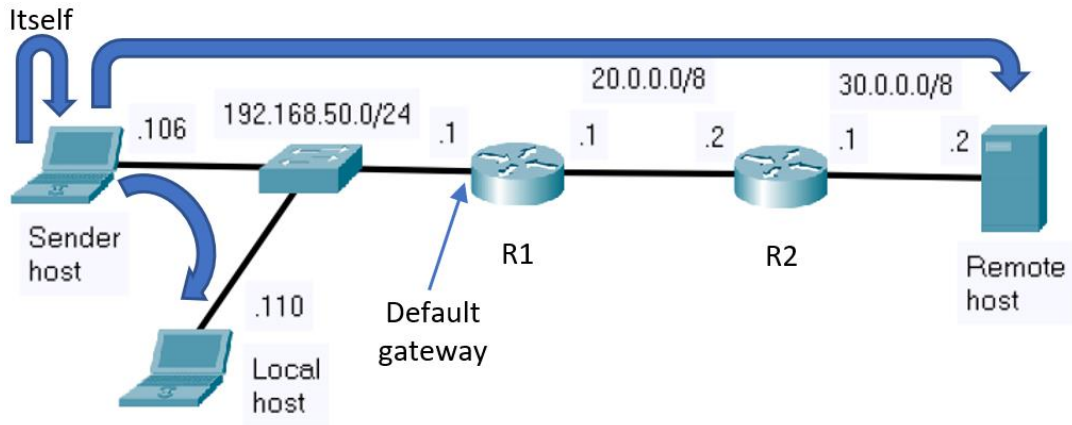


Figure 6.3 Possible packet destinations when a host is transmitting data

When a host is configured with IPv4 address (Figure 6.4), subnet mask and default gateway, it updates its routing table accordingly (Figure 6.5).

```
C:\Users\Admin>ipconfig

IPv4 Address. . . . . : 192.168.50.106
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.50.1
```

Figure 6.4 Host IP configuration

```
C:\Users\Admin>route print

IPv4 Route Table
=====
Active Routes:
Network Destination    Netmask          Gateway          Interface        Metric
0.0.0.0                0.0.0.0          192.168.50.1    192.168.50.106   55
127.0.0.0              255.0.0.0        On-link         127.0.0.1        331
127.0.0.1              255.255.255.255 On-link         127.0.0.1        331
127.255.255.255        255.255.255.255 On-link         127.0.0.1        331
192.168.50.0           255.255.255.0    On-link         192.168.50.106   311
192.168.50.106         255.255.255.255 On-link         192.168.50.106   311
192.168.50.255        255.255.255.255 On-link         192.168.50.106   311
224.0.0.0              240.0.0.0        On-link         127.0.0.1        331
224.0.0.0              240.0.0.0        On-link         192.168.50.106   311
255.255.255.255        255.255.255.255 On-link         127.0.0.1        331
255.255.255.255        255.255.255.255 On-link         192.168.50.106   311
=====
```

Figure 6.5 Host routing table

If a host is sending a packet to a device that is configured with the same network IP as the host device, the packet is forwarded out of the host interface, through the intermediary device, and to the destination device directly. Consider the network below (Figure 6.6), the packet is

transmitted from 192.168.50.106 to 192.168.50.110. The best (longest) matching route entry (Figure 6.7) in this case is marked with red.

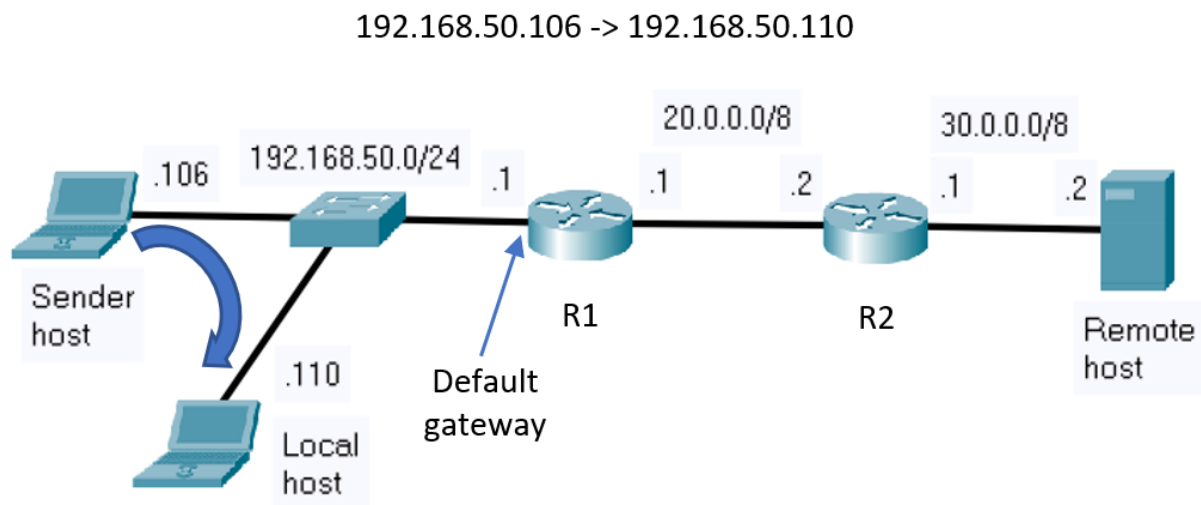


Figure 6.6 Sending host transmitting to a local host

```
C:\Users\Admin>route print

IPv4 Route Table
=====
Active Routes:
Network Destination        Netmask          Gateway          Interface        Metric
-----
0.0.0.0                    0.0.0.0         192.168.50.1    192.168.50.106   55
127.0.0.0                  255.0.0.0       On-link         127.0.0.1        331
127.0.0.1                  255.255.255.255 On-link         127.0.0.1        331
127.255.255.255           255.255.255.255 On-link         127.0.0.1        331
192.168.50.0              255.255.255.0   On-link         192.168.50.106   311
192.168.50.106            255.255.255.255 On-link         192.168.50.106   311
192.168.50.255           255.255.255.255 On-link         192.168.50.106   311
224.0.0.0                  240.0.0.0       On-link         127.0.0.1        331
224.0.0.0                  240.0.0.0       On-link         192.168.50.106   311
255.255.255.255           255.255.255.255 On-link         127.0.0.1        331
255.255.255.255           255.255.255.255 On-link         192.168.50.106   311
=====
```

Figure 6.7 The best matching route entry for a local host

If a host is sending a packet to a remote host, the packet is forwarded out of the host interface, through the intermediary device, and to the gateway. Consider the network below (Figure 6.8), the packet is transmitted from 192.168.50.106 to 30.0.0.2. The best (longest) matching route entry (Figure 6.9) in this case is the default route, marked with red.

When the packet arrives on the interface of a router, the router de-encapsulates the Layer 2 header and trailer. Then, it examines the destination IPv4 address of the packet and searches for the best match in its IPv4 routing table. When the best match in the routing table is found, the router forwards the packet according to the information in the route entry, encapsulating it

in the new Layer 2 header and trailer. In our example, the best (longest) matching route entry (Figure 6.10) is a static route, marked with red.

And the hop-by-hop forwarding process continues until the packet reaches the destination.

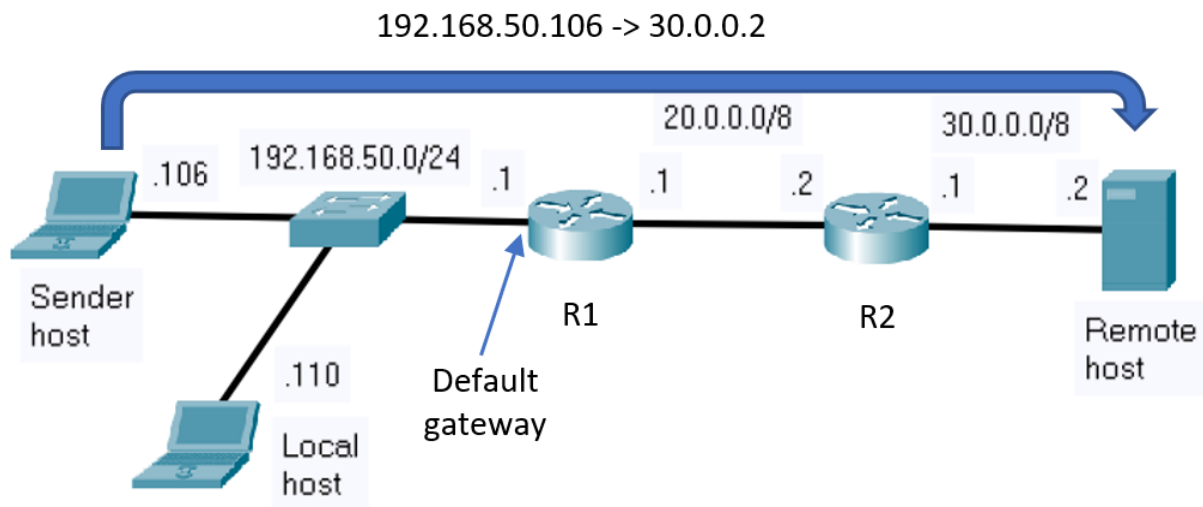


Figure 6.8 Sending host transmitting to a remote host

```
C:\Users\Admin>route print

IPv4 Route Table
=====
Active Routes:
Network Destination        Netmask          Gateway          Interface        Metric
-----
0.0.0.0                    0.0.0.0         192.168.50.1    192.168.50.106   55
127.0.0.0                  255.0.0.0       On-link         127.0.0.1        331
127.0.0.1                  255.255.255.255 On-link         127.0.0.1        331
127.255.255.255           255.255.255.255 On-link         127.0.0.1        331
192.168.50.0              255.255.255.0   On-link         192.168.50.106   311
192.168.50.106            255.255.255.255 On-link         192.168.50.106   311
192.168.50.255           255.255.255.255 On-link         192.168.50.106   311
224.0.0.0                  240.0.0.0       On-link         127.0.0.1        331
224.0.0.0                  240.0.0.0       On-link         192.168.50.106   311
255.255.255.255           255.255.255.255 On-link         127.0.0.1        331
255.255.255.255           255.255.255.255 On-link         192.168.50.106   311
=====
```

Figure 6.9 The best matching route entry for a remote host

```

R1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

C    20.0.0.0/8 is directly connected, FastEthernet1/0
S    30.0.0.0/8 [1/0] via 20.0.0.2
C    192.168.50.0/24 is directly connected, FastEthernet0/0

```

Figure 6.10 The best matching route entry for a remote host in R1 routing table

When a router is configured with an IPv4 address and a subnet mask, it updates its routing table accordingly. Additionally, a router can learn about remote networks in one of two ways: manually and dynamically. In the first case, the remote networks are manually entered into the routing table using static routes. In the second case, remote networks are automatically learned using a dynamic routing protocol.

In the previous example, R1 router was manually configured with a static route to reach the remote network address, 30.0.0.0/8, through (via) the IP address of the next hop router, 20.0.0.2 (Figure 6.11).

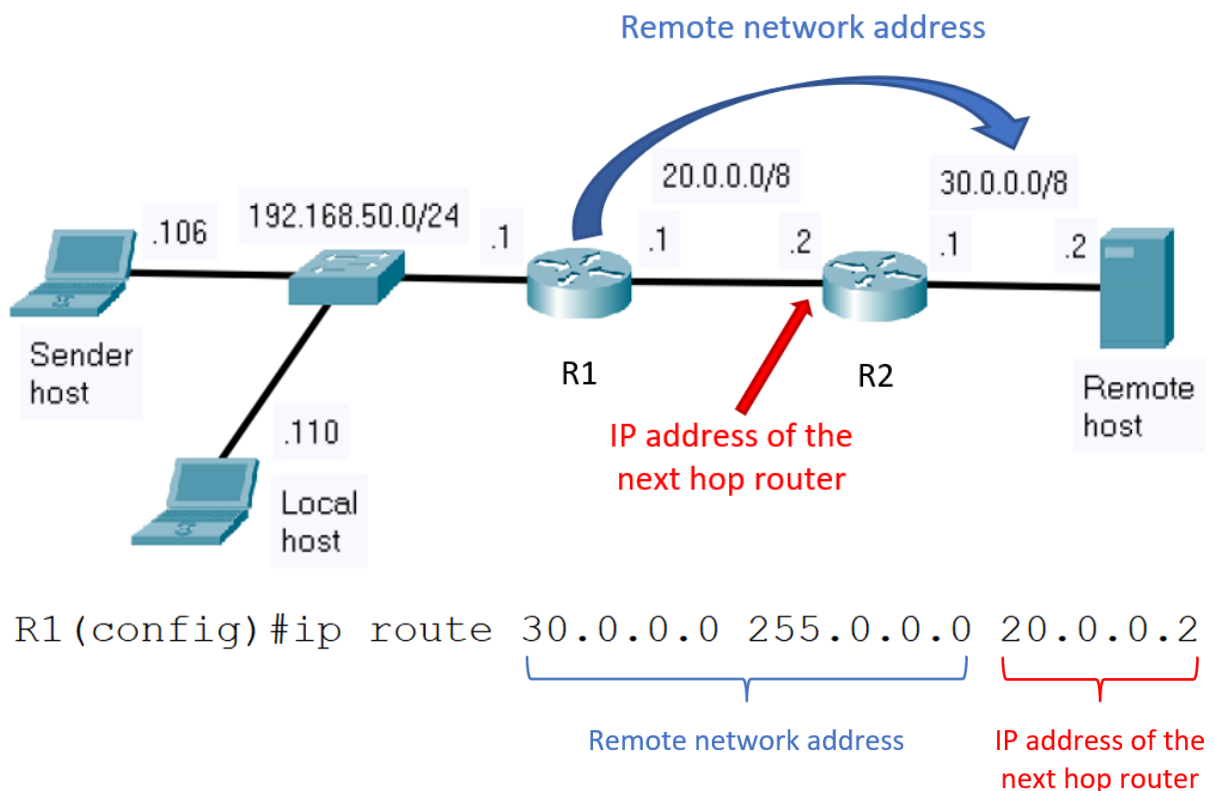


Figure 6.11 Static route configuration

2.2 Dynamic Host Configuration Protocol v4

Dynamic Host Configuration Protocol v4 (DHCPv4) assigns network configuration information dynamically. The IPv4 address is assigned, or leased, for a limited period of time. When the lease expires, the client must ask for another address. Usually, the server reassigns the same address to the client.

The DHCPv4 service can run on various devices such as a dedicated server or a router. The DHCP process starts when the client joins a network. The client sends a DHCPDISCOVER broadcast message to find the DHCPv4 server. The DHCPv4 server reserves an available IPv4 address to lease to the client and sends the binding DHCPOFFER message to the client. The client sends a DHCPREQUEST broadcast message as a binding acceptance notice. The server replies with a DHCPACK message (Figure 6.12).

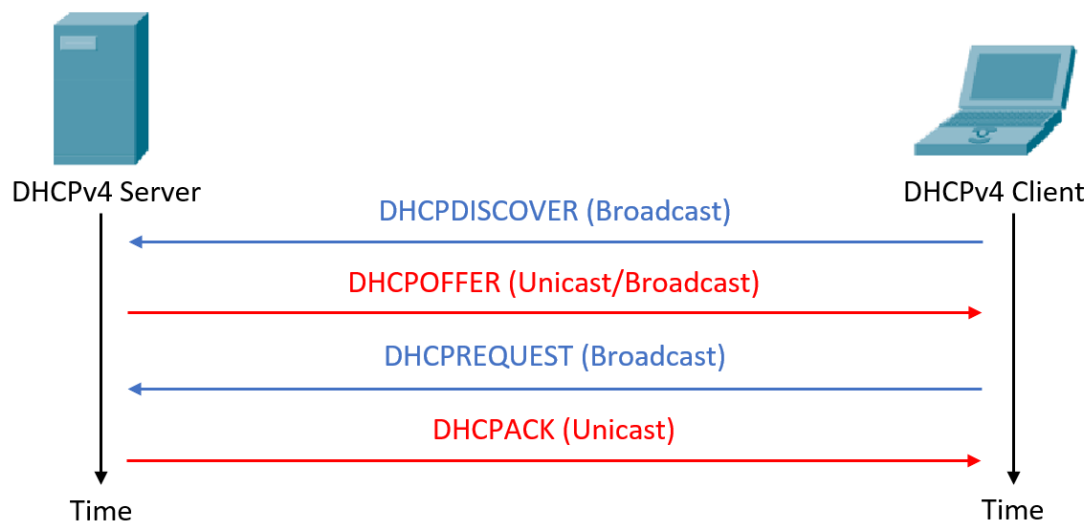


Figure 6.12 DHCP process

The previous sequence of operations can also be observed in Wireshark (Figure 6.13) when using the `ipconfig /release` and `ipconfig /renew` (on Windows OS) and `dhclient` (on Linux OS) commands:

No.	Time	Source	Destination	Protocol	Length	Info
171	10.925562	192.168.0.100	192.168.0.1	DHCP	342	DHCP Release - Transaction ID 0x9a7c44e2
411	17.861982	0.0.0.0	255.255.255.255	DHCP	342	DHCP Discover - Transaction ID 0x31d82096
412	17.866707	192.168.0.1	192.168.0.100	DHCP	590	DHCP Offer - Transaction ID 0x31d82096
413	17.869187	0.0.0.0	255.255.255.255	DHCP	366	DHCP Request - Transaction ID 0x31d82096
427	18.381342	192.168.0.1	192.168.0.100	DHCP	590	DHCP ACK - Transaction ID 0x31d82096

Figure 6.13 Wireshark capture of DHCP process

3. Practical activity

3.1 Discuss the theoretical aspects presented in this chapter.

3.2 Visualize the DHCP network packets that are delivered by your local machine. For this do the following:

- Start a Wireshark capture
- Apply the “dhcp” packet filter in the Wireshark window
- Open a command prompt (terminal)
- Run the following commands: **ipconfig /release** and **ipconfig /renew** (on Windows OS) and **dhclient** (on Linux OS)
- Inspect the Wireshark capture which shows the DHCP sequence of operations
- In the Wireshark capture identify the broadcast address in the Layer 2 encapsulation (Ethernet encapsulation)

3.3 Consider the network topology below (Figure 6.14):

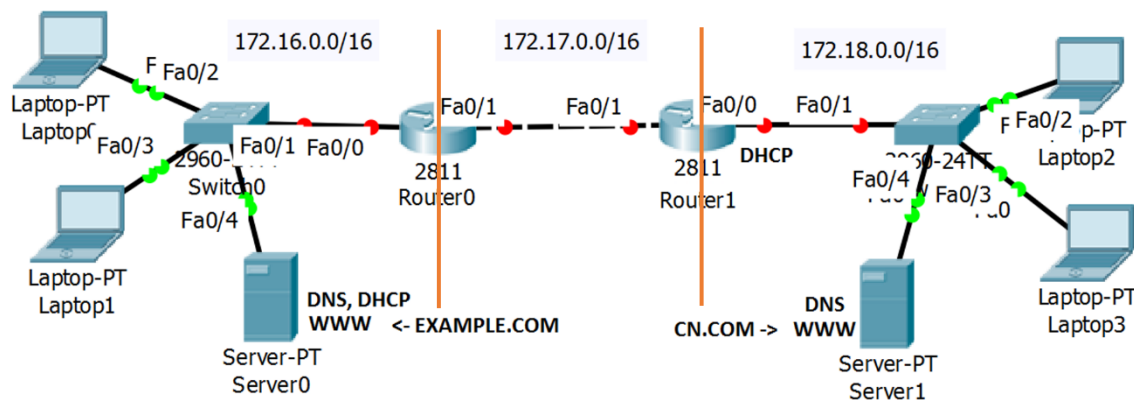


Figure 6.14 Test network topology

Step 1: Before configuring the network devices, assign a unique IP address and the corresponding subnet mask to each network interface and fill in the Table 6.1:

Table 6.1 IPv4 addresses and netmasks for the test network

Device	Interface	IP Address	Subnet mask	Default gateway
Laptop0	Fa	___ . ___ . ___ . ___	___ . ___ . ___ . ___	___ . ___ . ___ . ___
Laptop1	Fa	___ . ___ . ___ . ___	___ . ___ . ___ . ___	___ . ___ . ___ . ___
Server0	Fa	___ . ___ . ___ . ___	___ . ___ . ___ . ___	___ . ___ . ___ . ___
Router0	Fa0/0	___ . ___ . ___ . ___	___ . ___ . ___ . ___	-
Router0	Fa0/1	___ . ___ . ___ . ___	___ . ___ . ___ . ___	-
Router1	Fa0/1	___ . ___ . ___ . ___	___ . ___ . ___ . ___	-
Router1	Fa0/0	___ . ___ . ___ . ___	___ . ___ . ___ . ___	-
Laptop2	Fa	assigned by DHCP	assigned by DHCP	assigned by DHCP
Laptop3	Fa	assigned by DHCP	assigned by DHCP	assigned by DHCP
Server1	Fa	___ . ___ . ___ . ___	___ . ___ . ___ . ___	___ . ___ . ___ . ___

Note*: pay attention to the interface names of the router you are using, some routers may only have GigabitEthernet interfaces.

Step 2: Assign static IPv4 addresses to router interfaces

```

Router1>enable
Router1#configure terminal
Router1(config)#interface _____
Router1(config-if)#ip address _____
Router1(config-if)#no shutdown
Router1(config-if)#exit

```

Step 3: Configure DHCP

The Table 6.2 contains the configuration steps and the corresponding commands to configure the DHCP functionality on a Cisco router.

Table 6.2 DHCP functionality configuration on a Cisco router

No	Operation	Command	Example
1	Exclude IP addresses	<i>Router(config)#ip dhcp excluded-address start_address end_address</i>	<i>ip dhcp excluded-address 172.18.0.1 172.18.0.5</i>
2	Configure the pool name	<i>Router(config)# ip dhcp pool name</i>	<i>ip dhcp pool TestDHCP</i>
3	Configure the addresses (specifying the network address and the mask to be used)	<i>Router(dhcp-config)# network network-number [mask/prefix-length]</i>	<i>network 172.18.0.0 255.255.0.0</i>
4	Configure the Default Router (the Gateway) for the Clients	<i>Router(dhcp-config)# default-router address</i>	<i>default-router 172.18.0.1</i>
5	Set up the IP Domain Name System Servers for the Clients	<i>Router(dhcp-config)# dns-server address [address2 ...address5]</i>	<i>dns-server 8.8.8.8</i>
6	Visualize the DHCP pools information and DHCP address bindings	<i>Router#show ip dhcp pool</i>	<i>show ip dhcp binding</i>

Step 4: Set the static routes

Step 4.1: Identify the static route needed on each router and fill in the Table 6.3:

Table 6.3 Static routes for the test network

Device	Destination network	Destination mask	Next hop
Router0	___ . ___ . ___ . ___	___ . ___ . ___ . ___	___ . ___ . ___ . ___
Router1	___ . ___ . ___ . ___	___ . ___ . ___ . ___	___ . ___ . ___ . ___

Step 4.2: Configure the static routes on the Cisco routers

General syntax:

Router(config)#ip route netw_dest_address next_hop_address/interface

Example: *Router0(config)#ip route 172.18.0.0 255.255.0.0 172.17.0.2*

Note*: use your own IP addresses and mask when configuring the devices, not the one provided in this example

Visualize the routing table:

Router1#show ip route

Step 5: Test the connectivity between end devices from opposite networks.

a. *ping <target IP>*

b. *tracert <target IP>*

Optional step: Configure DHCP on Server0 also, similar with the example on Figure 6.15 (replace the IP address in the example with your IP addresses)

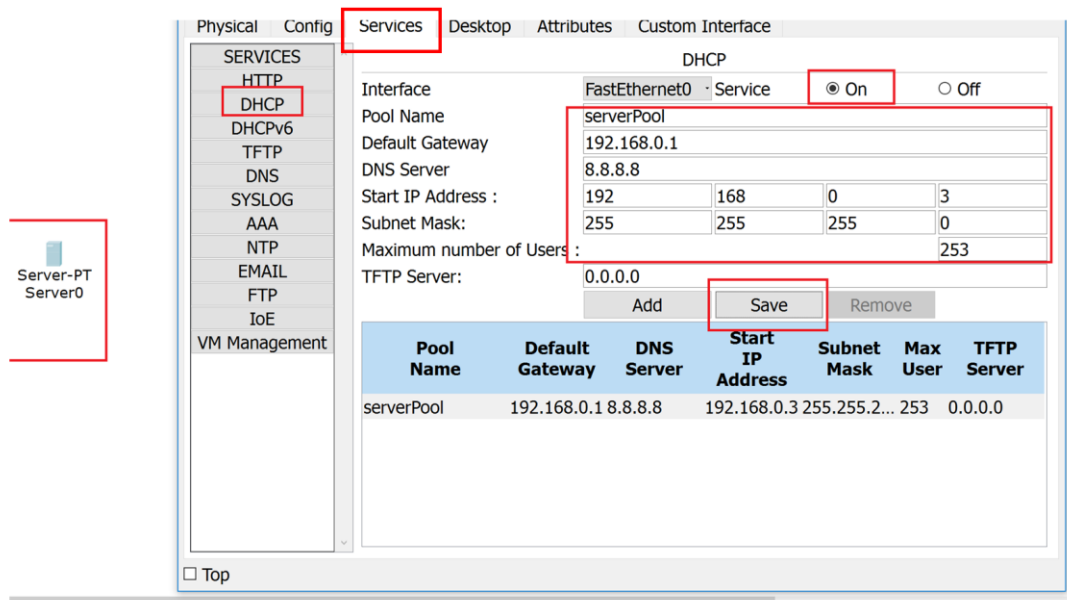


Figure 6.15 DHCP configuration example

CHAPTER 7: NETWORK LAYER – IPv6

1. Objectives

At the end of the practical activity, students will be able: to explain the characteristics of the IPv6 protocol, to describe the dynamic IPv6 configuration, to explain the routing process, and to implement basic IPv6 network configurations.

2. Theoretical considerations

The current practical work focuses on the Network layer of the ISO/OSI stack (Figure 7.1).

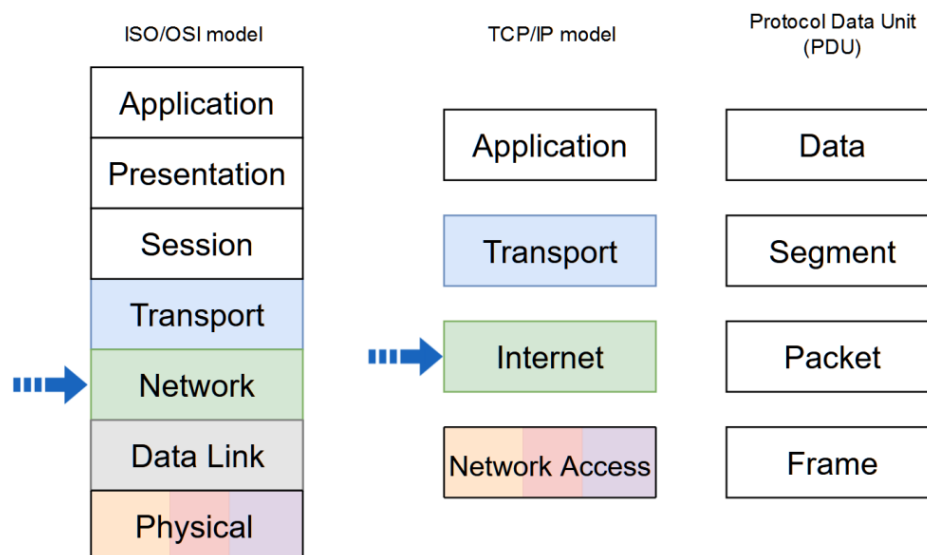


Figure 7.1 Network stack models and PDU naming in each level. The arrows indicate the addressed layers in the current activity

2.1 IPv6

IPv6 was developed by the Internet Engineering Task Force (IETF) to overcome the limitations of IPv4.

The main limitation of IPv4 is the exhaustion of addresses because the address request is larger than the address space provided by the 32 bits of the address. The solution for the IPv4 address depletion is private addressing and NAT. This solution in turn creates several drawbacks such as lack of end-to-end connectivity and increased network complexity.

IPv6 provides the following improvements:

- Increased address space based on 128 bit address;
- Improved packet handling due to the simplified header with fewer fields;
- Eliminates the need for NAT by eliminating the need for private addresses.

The packet header is presented in Figure 7.2:

Octet	0				1				2				3																			
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	Version				Traffic class				Flow label																							
32	Payload length								Next header								Hop limit															
64	Source IP Address																															
96																																
128																																
160																																
192	Destination IP Address																															
224																																
256																																
288																																

Figure 7.2 IPv6 packet header

- Version - version field, equal to 6;
- Traffic class - equivalent to DiffServ – DS field;
- Payload length – indicates the length of the payload of the IPv6 packet;
- Next header – defines the next level protocol;
- Hop limit – replaces the Time to live field in IPv4;
- Source address – the IPv4 address of the sender of the packet;
- Destination address – the IPv4 address of the receiver of the packet;

IPv6 packet may contain extension headers, placed between IPv6 header and the payload, providing optional network layer information. Routers do not fragment IPv6 packets.

IPv6 addresses are 128 bits in length. The preferred format for writing an IPv6 address is x:x:x:x:x:x:x, with each “x” consisting of four hexadecimal values, 4 bits being represented by a hexadecimal digit. Hextet is an unofficial term, it refers to a segment of 16 bits (4 values in hexadecimal). Figure (Figure 7.3) shows an example of an IPv6 addresses in the preferred format:

Type	Format
Preferred	2001:0b20:0000:00d7:0000:0000:0000:0012

Figure 7.3 IPv6 address format for writing – preferred

There are two rules to reduce or compress IPv6 representation. The first rule is to omit the zeros that are at the beginning of each hextet - leading 0s (zeros) (Figure 7.4).

Type	Format
Preferred	2001:0b20:0000:00d7:0000:0000:0000:0012
No leading 0s	2001:b20:0:d7:0:0:0:12

Figure 7.4 IPv6 address format for writing – no leading 0s

The second rule is to omit the segments (hextets) that contain all the bits 0 and replace them with "double colon" (::). This replacement can be done only once (Figure 7.5).

Type	Format
Preferred	2001:0b20:0000:00d7:0000:0000:0000:0012
No leading 0s	2001:b20:0:d7:0:0:0:12
Compressed	2001:b20:0:d7::12
or	
Compressed	2001:b20::d7:0:0:0:12

Figure 7.4 IPv6 address format for writing – compressed

Types of IPv6 addresses:

- Unicast
 - Uniquely identifies an interface
 - The source address must be unicast
- Multicast
 - It is used to send a single IPv6 packet to multiple destinations
 - IPv6 does not have a broadcast address, but there is a multicast address that provides the same result
 - Well-Known Multicast Addresses
 - ff02 :: 1: All IPv6 devices
 - ff02 :: 2: All IPv6 routers
 - ff02 :: 5: All OSPFv3 routers
 - ff02 :: a: All EIGRP (IPv6) routers
- Anycast
 - Any unicast address that can be assigned to multiple devices
 - A packet sent to anycast address is routed to the nearest device with that address

IPv6 prefix length indicates the network portion of an IPv6 address. It is represented in slash notation and can range from 0 to 128. The recommended IPv6 prefix length for LANs is /64. Figure 7.5 shows an example of an IPv6 address and prefix length: 2001:b20:0:d7::12/64.

Prefix (64 bits)	Interface ID (64 bits)
2001:0b20:0000:00d7	0000:0000:0000:0012

Figure 7.5 IPv6 address and prefix length example

Types of unicast addresses (Figure 7.6):

- Global Unicast Address (GUA)
 - Globally unique
 - Routable on the Internet
 - Similar to a public IPv4 address
- Link-local Address (LLA)
 - Required for every IPv6-enabled device
 - Created even if the device has not been assigned a global unicast address
 - For communication with other devices from the same local link
 - Allow devices to communicate only on the same link
 - Unique only in the local link

- Not routable on the Internet
- They are in the range FE80::/10
- The router's link-local address is usually used as the default gateway
- Unique Local Address (ULA)
 - Local addressing within a site or between a limited number of sites

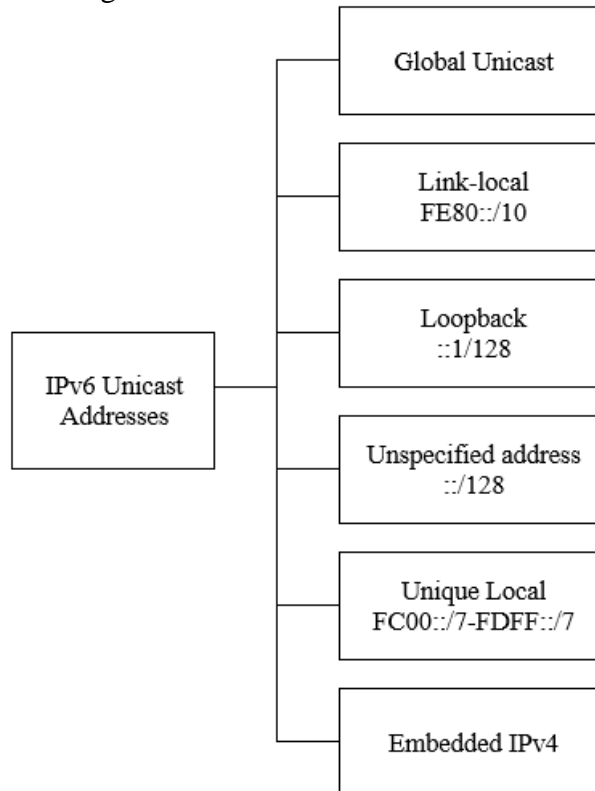


Figure 7.6 Types of unicast addresses

- Structure of the Global Unicast Addresses (GUA):
- Global routing prefix
 - Network
 - Portion of the address assigned by the provider
 - Typical /48
- Subnet ID
 - For subnetting in an organization
 - Usually, 16 bits
- Interface ID
 - The equivalent of the host portion of an IPv4 address
 - Usually, 64 bits

Figure 7.7 shows an example of an IPv6 global unicast address: 2001:b20:0:d7::12/64

Global Routing Prefix	Subnet ID	Interface ID
2001:0b20:0000	00d7	0000:0000:0000:0012

Figure 7.7 IPv6 global unicast address example

2.2 Host configuration

Methods:

- Static
 - Manual configuration of the IPv6 address
- Dynamic
 - Stateless Address Autoconfiguration (SLAAC)
 - Stateful DHCPv6

A device obtains the IPv6 addressing information dynamically, through Internet Control Message Protocol version 6 (ICMPv6) messages. IPv6 routers periodically send out ICMPv6 Router Advertisement (RA) messages to all IPv6-enabled devices on the network. An RA message will also be sent in response to a host sending an ICMPv6 Router Solicitation (RS) message, which is a request for an RA message.

The ICMPv6 RA message is a suggestion to devices on how to obtain IPv6 addressing information. The ICMPv6 RA message includes the following:

- Network prefix and prefix length
- Default gateway address
- DNS addresses and domain name

There are three methods for RA messages:

- Method 1: SLAAC - prefix, prefix length, and default gateway address
- Method 2: SLAAC with a stateless DHCPv6 server – partial information, the rest of the information, such as DNS addresses, needs to be obtained from a stateless DHCPv6 server
- Method 3: Stateful DHCPv6 (no SLAAC) - default gateway address, the rest of the information, needs to be obtained from a stateful DHCPv6 server

The decision of how a client will obtain IPv6 addressing information depends on the settings within the RA message. An ICMPv6 RA message includes three flags to identify the dynamic options available to a host, as follows:

- A flag - Address Autoconfiguration flag. Use Stateless Address Autoconfiguration (SLAAC) to create an IPv6 GUA.
- O flag - Other Configuration flag. Other information is available from a stateless DHCPv6 server.
- M flag - Managed Address Configuration flag. Use a stateful DHCPv6 server to obtain an IPv6 GUA.

- Method 1 – SLAAC (Figure 7.8)

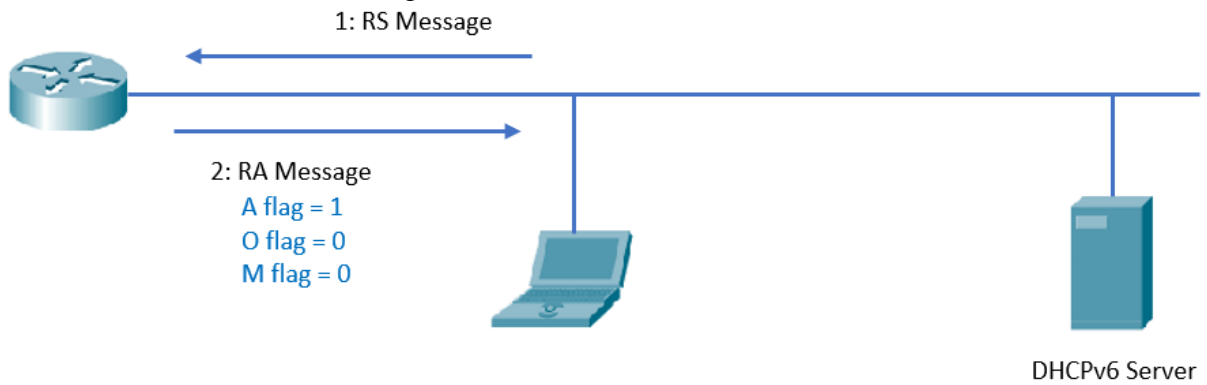


Figure 7.8 SLAAC process

- Method 2 - SLAAC with a stateless DHCPv6 server (Figure 7.9)

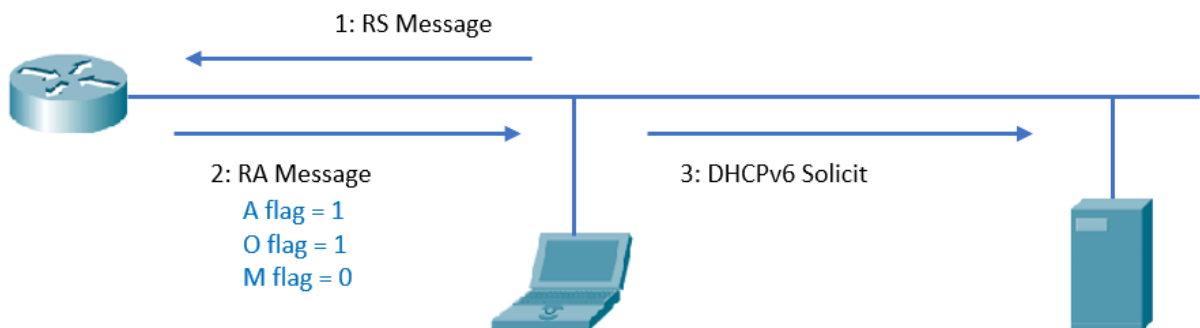


Figure 7.9 SLAAC with a stateless DHCPv6 server process

- Method 3 - Stateful DHCPv6 (no SLAAC) (Figure 7.10)

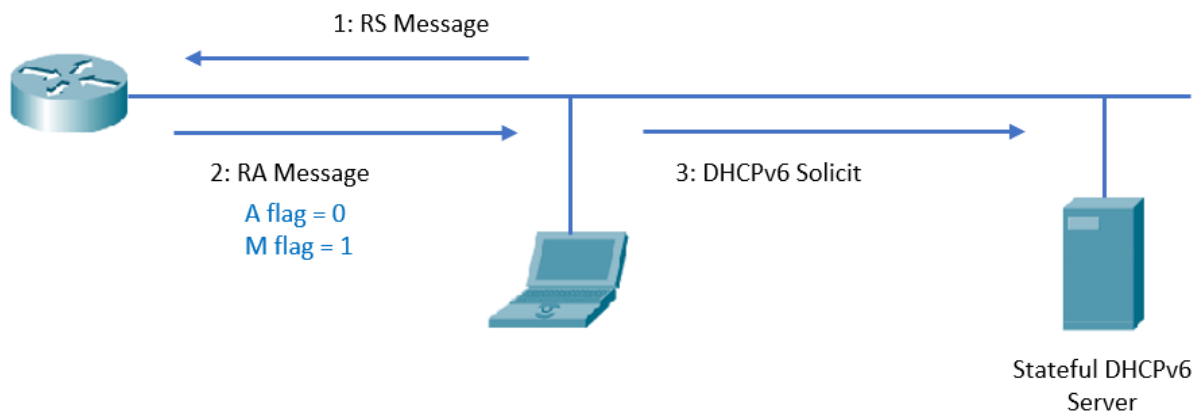


Figure 7.10 Stateful DHCPv6 process

3. Practical activity

3.1 Discuss the theoretical aspects presented in this chapter.

3.2 Consider the network topology below (Figure 7.11):

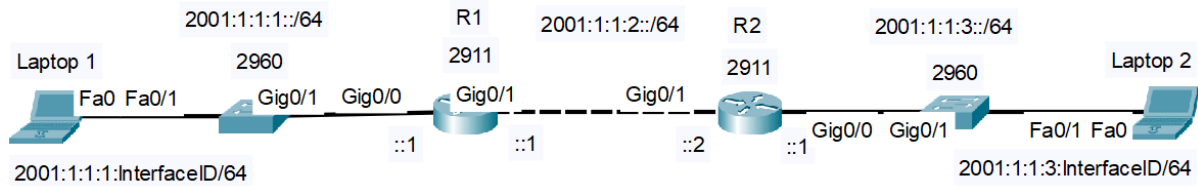


Figure 7.11 Test network topology

Step 1: Before configuring the network devices, discuss the IPv6 address assignment in the Table 7.1:

Table 7.1 IPv6 addresses for the test network

Device	Interface	IPv6 Address
Laptop 1	Fa0	DHCPv6
Laptop 2	Fa0	DHCPv6
R1	Gig0/0	2001:1:1:1::1/64 fe80::1 link-local
R1	Gig0/1	2001:1:1:2::1/64
R2	Gig0/1	2001:1:1:2::2/64 fe80::2 link-local
R2	Gig0/0	2001:1:1:3::1/64

Note*: pay attention to the interface names of the router you are using, some routers may only have FastEthernet interfaces.

Step 2: Assign hostnames, enable IPv6 routing and assign static IPv6 addresses to router interfaces.

Example:

```
R1>enable
R1#configure terminal
Router(config)#hostname R1
R1(config)#ipv6 unicast-routing
```

```
R1(config)# interface gigabitEthernet 0/0
R1(config-if)#ipv6 address fe80::1 link-local
R1(config-if)#ipv6 address 2001:1:1:1::1/64
R1(config-if)#no shutdown
```

Use the following command to display the IPv6 addresses configured on the router:

```
R1#show ipv6 interface brief
```

Step 3: Configure a static route on each router pointed to the IPv6 address of Gig0/1 on the other router. For R1 router specify the LLA address for the next hop and for the R2 router specify the GUA address the next hop. Discuss the differences!

```
R1(config)#ipv6 route 2001:1:1:3::/64 GigabitEthernet0/1 FE80::2
```

```
R2(config)# ipv6 route 2001:1:1:1::/64 2001:1:1:2::1
```

Use the following command to display the IPv6 routing table:

```
Router#show ipv6 route
```

Step 4: Verify SLAAC Address Assignment (Figures 7.12 and 7.13).

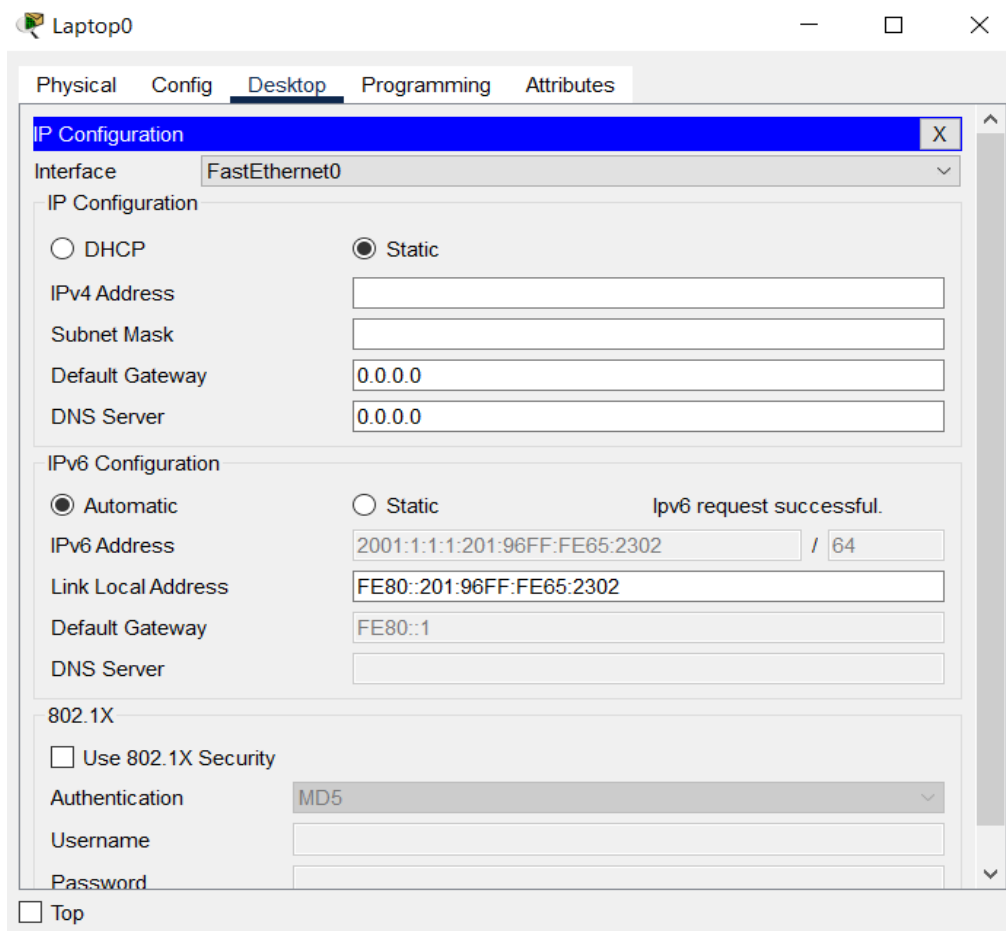


Figure 7.12 IP configuration view - GUI

```

C:\>ipconfig /all

FastEthernet0 Connection:(default port)

    Connection-specific DNS Suffix...:
    Physical Address.....: 0001.9665.2302
    Link-local IPv6 Address.....: FE80::201:96FF:FE65:2302
    IPv6 Address.....: 2001:1:1:1:201:96FF:FE65:2302
    IPv4 Address.....: 0.0.0.0
    Subnet Mask.....: 0.0.0.0
    Default Gateway.....: FE80::1
                        0.0.0.0
    DHCP Servers.....: 0.0.0.0
    DHCPv6 IAID.....:
    DHCPv6 Client DUID.....: 00-01-00-01-23-BA-17-8B-00-01-96-65-23-02
    DNS Servers.....: ::
                        0.0.0.0

```

Figure 7.13 IP configuration view - CLI

Step 5: Test the connectivity between end devices from opposite networks (Figure 7.14).

- a. `ping <target IP>`
- b. `tracert <target IP>`

```

C:\>ping 2001:1:1:3:2D0:BAFF:FE66:228A

Pinging 2001:1:1:3:2D0:BAFF:FE66:228A with 32 bytes of data:

Reply from 2001:1:1:3:2D0:BAFF:FE66:228A: bytes=32 time<1ms TTL=126
Reply from 2001:1:1:3:2D0:BAFF:FE66:228A: bytes=32 time<1ms TTL=126
Reply from 2001:1:1:3:2D0:BAFF:FE66:228A: bytes=32 time<1ms TTL=126
Reply from 2001:1:1:3:2D0:BAFF:FE66:228A: bytes=32 time<1ms TTL=126

Ping statistics for 2001:1:1:3:2D0:BAFF:FE66:228A:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>tracert 2001:1:1:3:2D0:BAFF:FE66:228A

Tracing route to 2001:1:1:3:2D0:BAFF:FE66:228A over a maximum of 30 hops:

  1  0 ms      0 ms      0 ms      2001:1:1:1::1
  2  0 ms      0 ms      0 ms      2001:1:1:2::2
  3  0 ms      10 ms     0 ms      2001:1:1:3:2D0:BAFF:FE66:228A

Trace complete.

```

Figure 7.14 Connectivity testing commands

Step 6: Replace the configured static routes with default routes and test the connectivity between end devices from opposite networks. In IPv6, the default route is `::/0`.

```

R1(config)#no ipv6 route 2001:1:1:3::/64 GigabitEthernet0/1 FE80::2
R2(config)#no ipv6 route 2001:1:1:1::/64 2001:1:1:2::1

```

```
R1(config)#ipv6 route ::/0 _____
R2(config)#ipv6 route ::/0 _____
```

Step 7: Configure R1 to provide stateless DHCPv6 for Laptop 1.

```
R1(config)#ipv6 dhcp pool R1_NET1
R1(config-dhcpv6)#dns-server 2001:1:1:1::F
R1(config-dhcpv6)#domain-name NET1.com
R1(config-dhcpv6)#exit
R1(config)#interface gigabitEthernet 0/0
R1(config-if)#ipv6 nd other-config-flag
R1(config-if)#ipv6 dhcp server R1_NET1
```

Step 8: Verify stateless DHCPv6 Address Assignment (Figures 7.15 and 7.16).

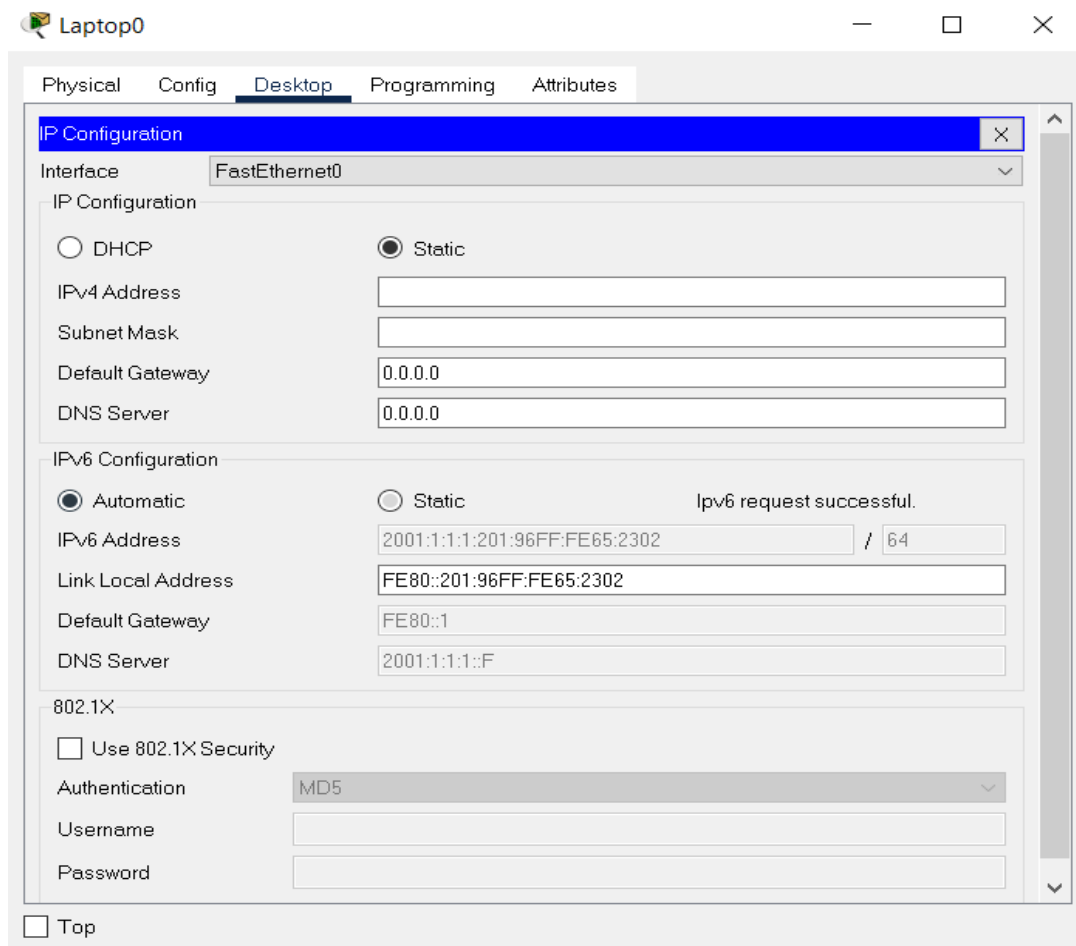


Figure 7.15 IP configuration view - GUI

```

C:\>ipconfig /all

FastEthernet0 Connection:(default port)

    Connection-specific DNS Suffix...: NET1.com
    Physical Address.....: 0001.9665.2302
    Link-local IPv6 Address.....: FE80::201:96FF:FE65:2302
    IPv6 Address.....: 2001:1:1:1:201:96FF:FE65:2302
    IPv4 Address.....: 0.0.0.0
    Subnet Mask.....: 0.0.0.0
    Default Gateway.....: FE80::1
                          0.0.0.0
    DHCP Servers.....: 0.0.0.0
    DHCPv6 IAID.....: 2061700019
    DHCPv6 Client DUID.....: 00-01-00-01-23-BA-17-8B-00-01-96-65-23-02
    DNS Servers.....: 2001:1:1:1::F
                          0.0.0.0

```

Figure 7.16 IP configuration view - CLI

Step 9: Configure R2 to provide stateful DHCPv6 for Laptop 2.

```

R2(config)#ipv6 dhcp pool R2_NET3
R2(config-dhcpv6)# address prefix 2001:1:1:3::/64
R2(config-dhcpv6)#dns-server 2001:1:1:3::A
R2(config-dhcpv6)#domain-name NET3.com
R2(config-dhcpv6)#exit
R2(config)#interface gigabitEthernet 0/0
R2(config-if)#ipv6 nd managed-config-flag
R2(config-if)#ipv6 dhcp server R2_NET3

```

Step 10: Verify stateful DHCPv6 Address Assignment (Figures 7.17 and 7.18).

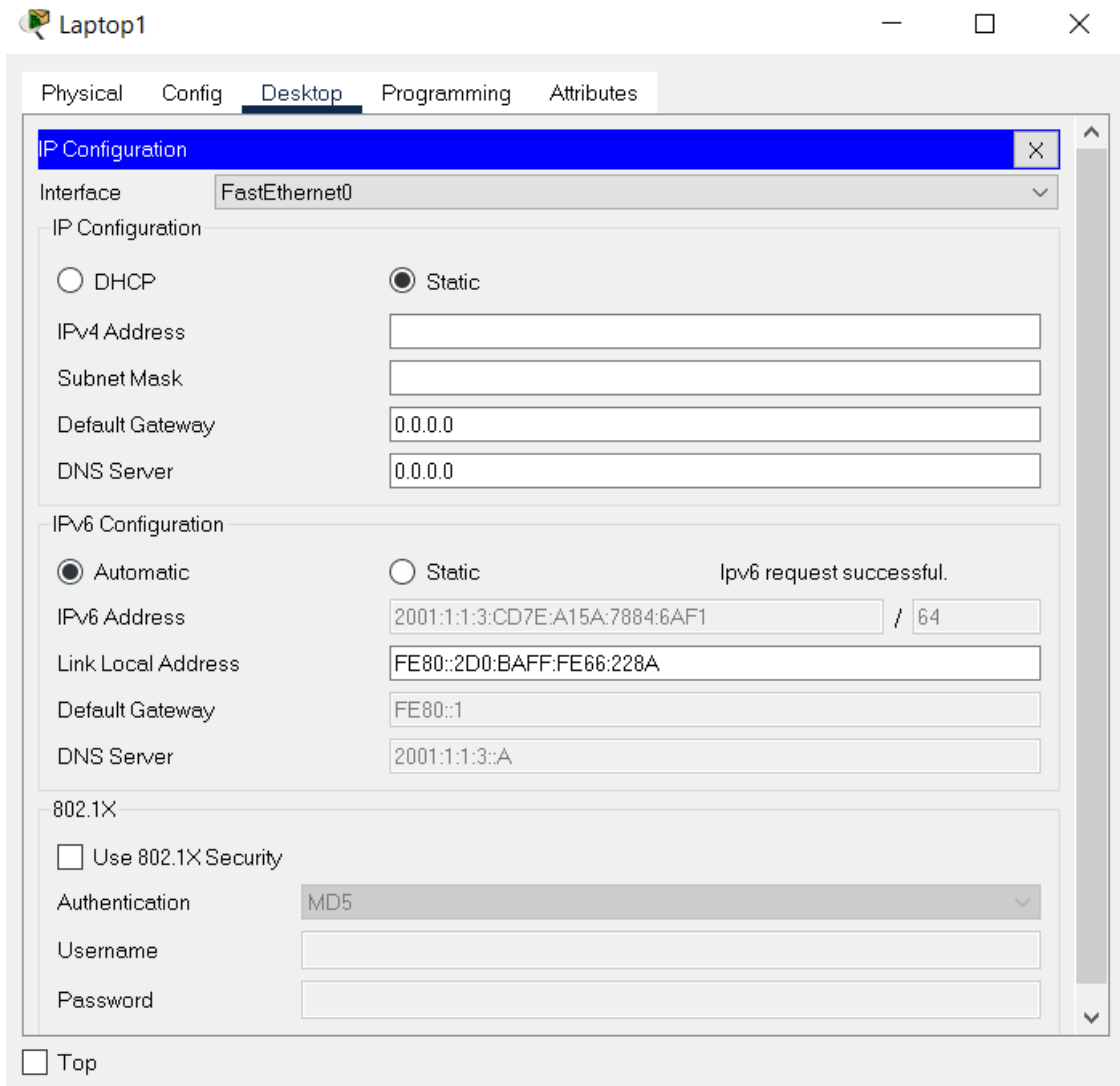


Figure 7.17 IP configuration view - GUI

```
C:\>ipconfig /all

FastEthernet0 Connection:(default port)

Connection-specific DNS Suffix...: NET3.com
Physical Address.....: 00D0.BA66.228A
Link-local IPv6 Address.....: FE80::2D0:BAFF:FE66:228A
IPv6 Address.....: 2001:1:1:3:CD7E:A15A:7884:6AF1
IPv4 Address.....: 0.0.0.0
Subnet Mask.....: 0.0.0.0
Default Gateway.....: FE80::1
                        0.0.0.0
DHCP Servers.....: 0.0.0.0
DHCPv6 IAID.....: 1998122365
DHCPv6 Client DUID.....: 00-01-00-01-40-4C-51-2B-00-D0-BA-66-22-8A
DNS Servers.....: 2001:1:1:3::A
                        0.0.0.0
```

Figure 7.18 IP configuration view - CLI

Step 11: Test the connectivity between end devices from opposite networks.

a. *ping* <target IP>

b. *tracert* <target IP>

CHAPTER 8: APPLICATION LAYER: NETWORK PROGRAMMING WITH SOCKETS

1. Objectives

Prerequisite: Use a working software environment for your preferred programming language (Java, C#, Python, C/C++, etc.)

At the end of the activity, students will be able to write software for socket applications and debug network applications using Wireshark.

2. Theoretical considerations

The current practical work focuses on the Transport and Application layers of the ISO/OSI stack (Figure 8.1).

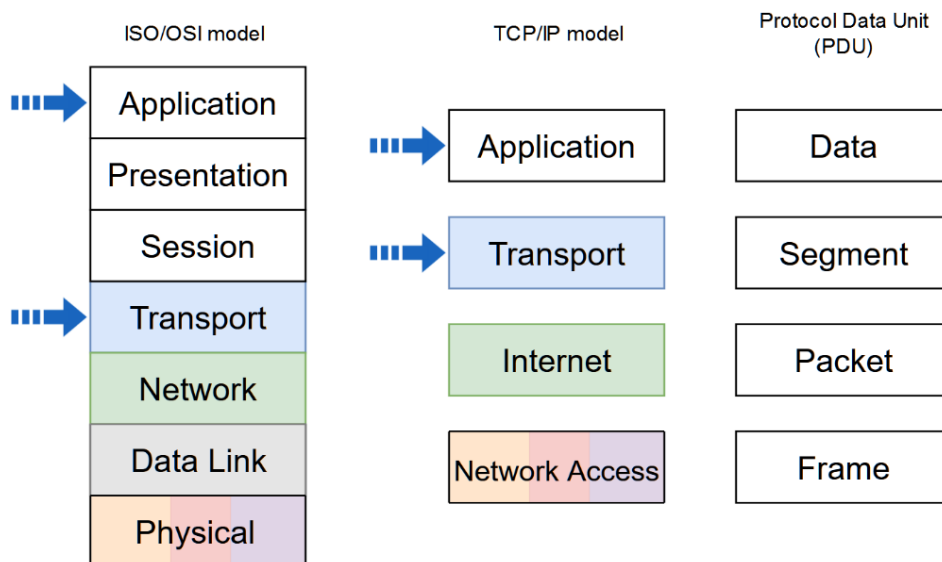


Figure 8.1 Network stack models and PDU naming in each level. The arrows indicate the addressed layers in the current activity

This practical activity addresses the programming side of software engineering and communication offered through the use of network sockets in a desktop environment. Socket programming is available in any high level programming language and sockets are transmitting information at the Application Layer. Sockets are used in different types of applications, such as: Client-Server, peer-2-peer systems, inter-process communication (on the same machine).

Network sockets can be constructed to use both IPv4 and IPv6 addresses. A socket is the combination of an IP address and a port number for use in a network application. A network application provides connectivity between different network devices. It is not possible to bind a socket to a port that is already in use by any other application, however the same port may be used concurrently by TCP and UDP transport layer protocols. The IP addresses identify the network device, but the port number uniquely identifies each running application on the current network device.

The operations that an application can perform on a socket are the following:

- **Create** - Creation of a socket object
- **Bind** - Configure the socket object to use a local pair of IP address and port number to accept connections
- **Listen** - Program the socket to wait for incoming connections
- **Accept** - Accept the incoming connection
- **Connect** - This operation is used by a client that wants to connect to a server
- **Send** - Used to send data over the socket to the remote destination
- **Receive** - Used to receive data which is sent from a remote location
- **Close** - close the connection between the two sockets

2.1. Working with sockets on the local machine

- In order to simulate a network on the local machine, the entire available loopback range: 127.0.0.0 - 127.255.255.255 can be used. The loopback network interface is available only on the local host and is mainly used for diagnostics and standalone network applications. Therefore, a simulated local network can use these IP addresses for communication. In order to test and confirm that this range can be used, a **ping** command can be run from the local terminal to verify connectivity to said IP addresses (Figure 8.2):

```
C:\Users\admin>ping 127.0.0.1

Pinging 127.0.0.1 with 32 bytes of data:
Reply from 127.0.0.1: bytes=32 time<1ms TTL=64
Reply from 127.0.0.1: bytes=32 time<1ms TTL=64
Reply from 127.0.0.1: bytes=32 time<1ms TTL=64
Reply from 127.0.0.1: bytes=32 time<1ms TTL=64

Ping statistics for 127.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\admin>ping 127.0.0.2

Pinging 127.0.0.2 with 32 bytes of data:
Reply from 127.0.0.2: bytes=32 time<1ms TTL=64
Reply from 127.0.0.2: bytes=32 time<1ms TTL=64
Reply from 127.0.0.2: bytes=32 time<1ms TTL=64
Reply from 127.0.0.2: bytes=32 time<1ms TTL=64

Ping statistics for 127.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Figure 8.2 Loopback addresses testing

- It is also possible to assign multiple valid IP addresses on the local interface, but this has to be done manually by statically allocating IP addresses to the interface. In this case, running the **ipconfig** command would show all the IP addresses assigned to the same interface. See an example below (Figure 8.3):

```

Wireless LAN adapter Wi-Fi 2:

Connection-specific DNS Suffix . : 
Link-local IPv6 Address . . . . . : fe80::fce4:5011:a37b:f929%22
IPv4 Address. . . . . : 10.0.0.1
Subnet Mask . . . . . : 255.0.0.0
IPv4 Address. . . . . : 10.0.0.2
Subnet Mask . . . . . : 255.0.0.0
IPv4 Address. . . . . : 20.0.0.1
Subnet Mask . . . . . : 255.0.0.0
IPv4 Address. . . . . : 20.0.0.2
Subnet Mask . . . . . : 255.0.0.0
IPv4 Address. . . . . : 30.0.0.1
Subnet Mask . . . . . : 255.0.0.0
IPv4 Address. . . . . : 30.0.0.2
Subnet Mask . . . . . : 255.0.0.0
IPv4 Address. . . . . : 172.16.0.1
Subnet Mask . . . . . : 255.255.0.0
IPv4 Address. . . . . : 172.16.0.2
Subnet Mask . . . . . : 255.255.0.0
IPv4 Address. . . . . : 172.16.0.3
Subnet Mask . . . . . : 255.255.0.0
IPv4 Address. . . . . : 192.168.0.35
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : fe80::8f3:92ff:fe9b:5d34%22
    
```

Figure 8.3 IP configuration view - CLI

- After having assigned the IP addresses, sockets can now be created to use these IP addresses.

2.2. TCP Sockets

- TCP (Transmission Control Protocol) sockets are connection oriented and represent a reliable data transmission mechanism that allows data to be received and processed in the same order it was transmitted.
- The Figure 8.4 shows a Wireshark traffic capture on the “Adaptor for loopback traffic capture”. The screenshot shows a client-server communication via sockets using the loopback addresses. The applied filter is **tcp.port == 1234**. The server is bound to the 127.0.0.1 address and awaits connections on port number 1234 while the client binds on the 127.0.0.2 address sending a payload of 14 bytes to the server.

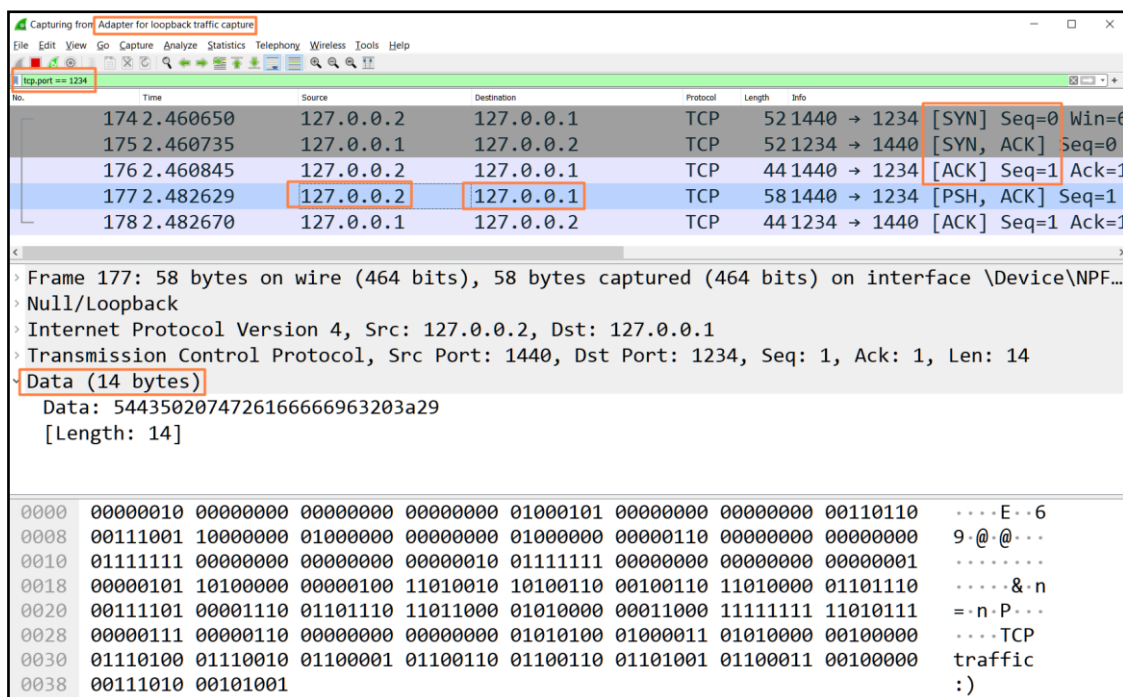


Figure 8.4 Wireshark capture of TCP socket communication

- The screenshot highlights the TCP mechanism represented by acknowledgement (ACK) packets. The first three packet exchanges (Figure 8.4) represent the 3-way handshake which is needed to establish the connection for any TCP connection (Figure 8.5) and following that the packet sending the payload is visible. This handshake assures that both hosts want to communicate and acknowledge the other host's intention to communicate.

3-way handshake

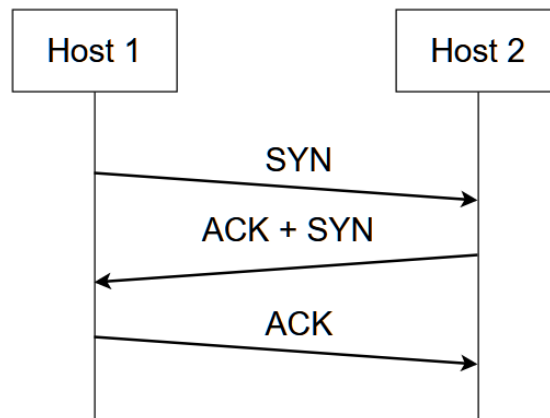


Figure 8.5 TCP 3-way handshake

- The Wireshark image shows a socket communication which remains open.
 - Answer the following question while working on the practical activity: If the socket connection is closed, what are the TCP flags that are set in order to close the connection?

2.3. UDP Sockets

- In contrast with the TCP sockets, UDP (User Datagram Protocol) sockets are not connection oriented and they do not provide reliable communication. This means they do not guarantee that network packets are delivered to the destination. The Figure 8.6 represents a Wireshark capture (again on the “Adaptor for loopback traffic capture”) of a UDP communication between two hosts. The applied filter is **udp.port == 1234**. As can be seen, there is no handshake performed and there aren't any ACK packets being transmitted.

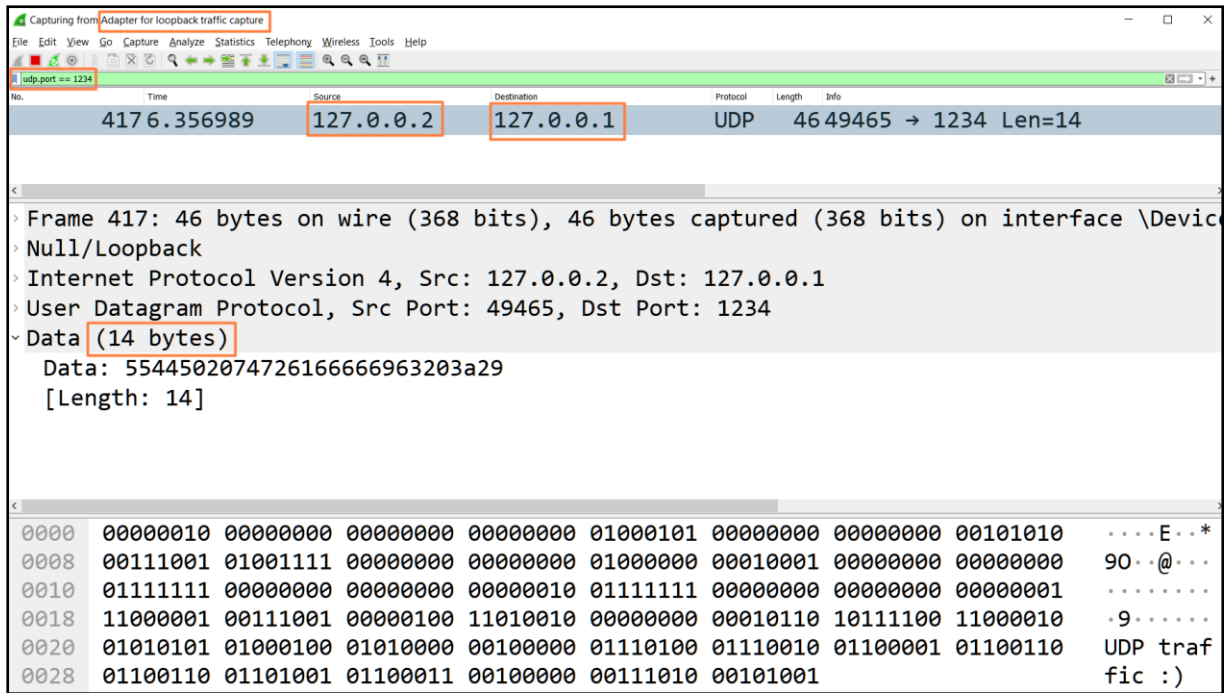


Figure 8.6 Wireshark capture of UDP socket communication

TCP and UDP socket communications are presented in Figure 8.7 a and b.

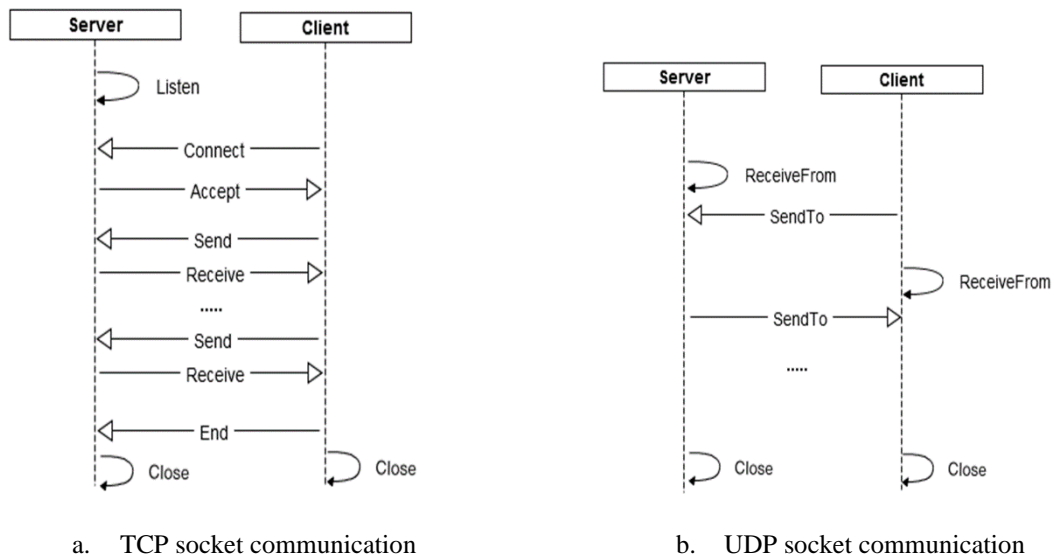


Figure 8.7 Socket communications

1. Implementation template

- A network device can function in 3 modes:
 - **Server:** Receiving device
 - **Client:** Sending device
 - **Relay:** Acting as an intermediary node in a communication and acts as both sending and receiving device. This type of network node can be encountered in Wireless Sensor Networks (WSN) where not all sensor nodes are in the wireless range of the collector device, therefore some nodes need to forward the information to the sink node (the collector node).

- This chapter provides a template for implementing the relay communication node using OOP concepts (the template is written in pseudocode, not in a particular programming language). This is not the only possibility to organize the code, students can choose any software design methodology they are comfortable with.

Relay node implementation template

```

class RelayNode {
public:
    RelayNode(IPAddress, serverPortNr) {
        m_server.listen(IPAddress, serverPortNr);
        m_client.bind(IPAddress);

        m_server.onReceive() => {
            ByteArray receivedData = m_server.readData();
            m_client.connectToHost(m_nextHopIpAddress, m_nextHopPortNr);
            m_client.sendData(receivedData);
            m_client.close();
        }
    }
    void setNextHopInformation(nextHopIpAddress, nextHopServerPortNr) {
        m_nextHopIpAddress = nextHopIpAddress;
        m_nextHopPortNr= nextHopServerPortNr;
    }

private:
    Server m_server; // server instance accepting connections
    Client m_client; // client instance sending data to the next hop
    IPAddress m_nextHopIpAddress; // next hop address used by the client instance
    int m_nextHopPortNr; // next hop port nr used by the client instance
}

void main() {
    RelayNode relay(127.0.0.1, 1234);
    relay.setNextHopInformation(127.0.0.2, 2345)
    ...
    // run application event loop
}

```

3. Practical activity

- Each student will be assigned one of the topologies below and the simulation scenario has to be implemented in software
- Besides the constraints imposed by each simulation scenario, the common tasks for each implementation are the following:
 - Use a programming language of choice to implement the network simulation
 - Use the loopback address range for addressing: 127.0.0.0 – 127.255.255.255
 - Test the implementation using Wireshark
 - Deliver the implementation (source code or link to online code versioning repository)

- Provide a Wireshark capture to prove the communication between different IP addresses
- Inspect the ratio between total delivered payload against the relevant application layer traffic / the ratio between the total packet length (headers and data) compared to the length of data sent (use Wireshark statistics or manual packet inspection)
- Depending on the implemented simulation, research the headers for TCP and/or UDP protocols. Using Wireshark, identify the header elements in the captured traffic

3.1 Ring communication

- Three computers are communicating in a single direction creating a loop (Figure 8.8)
- One of the computers initiates the communication sending the value '1'
- Upon receipt, each network device increments the received value and sends it to the next device
- The communication ends when the delivered payload reaches the value '100'
- Implementation hints:
 - Implement a single class which is instantiated 3 times with different communication parameters (reuse the code and do not duplicate it for each instance)
 - All communication uses TCP sockets (optional)

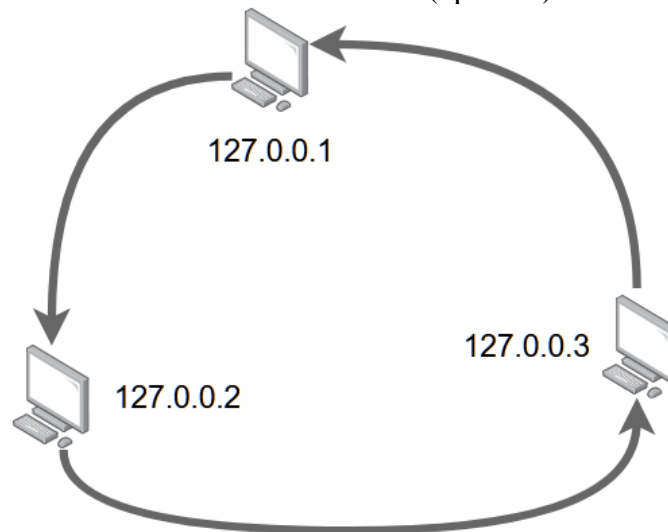


Figure 8.8 Ring communication network topology

3.2 Node selector

- There are three nodes in the topology: N1, N2, N3 (Figure 8.9)
- N1 increments a value 100 times and after every increment it sends the value to either N2 or N3 which are selected randomly for transmission
- When N2 receives an integer value which is a multiple of 3 it will send an ACK packet back to N1

- When N3 receives an integer value which is a multiple of 5 it will send an ACK packet back to N1
- Implementation hints:
 - Implement a single class for N2 and N3 which is instantiated with different communication parameters (reuse the code and do not duplicate it for each instance)
 - All communication uses UDP sockets (optional)

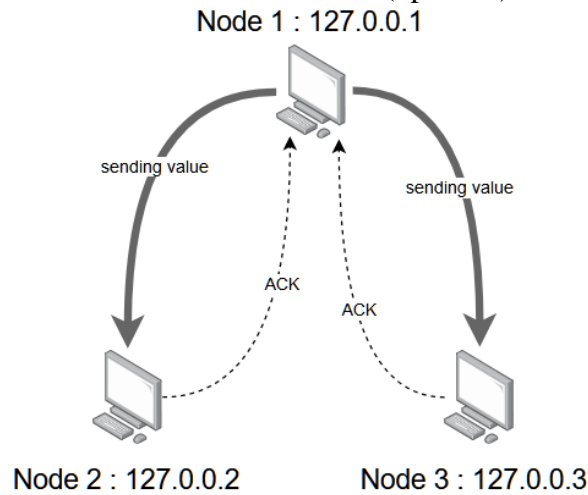


Figure 8.9 Node selector network topology

3.3 Relay nodes

- There are four nodes in the topology (Figure 8.10), Sender and three possible destinations (D1, D2, and D3)
- The Sender node is transmitting 100 packets containing an integer number randomly to one of the 3 possible destinations (D1, D2 or D3)
- After each packet transmission the integer number is incremented
- Every node can only send data to the next hop to which it is connected to, therefore a packet from the Sender to D3 must pass through D1 and D2

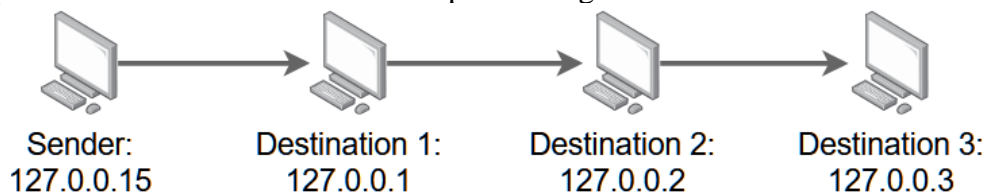


Figure 8.10 Relay nodes network topology

- Implementation hints:
 - The data payload that is transmitted via the socket has to contain the target IP address, so the payload has the following format (Figure 8.11):

Target IP address	Value
--------------------------	--------------

Figure 8.11 *Payload format*

- Every time a node receives a packet it verifies whether the received payload's target IP address is the same as the current node IP address. If it is identical, the communication stops here, otherwise the data is forwarded to the next hop.
- Implement a single class for D1, D2 and D3 which is instantiated with different communication parameters (reuse the code and do not duplicate it for each instance)

CHAPTER 9: ETHERNET, ARP AND NDP

1. Objectives

The objectives of this practical activity consist of understanding the structure of the Ethernet frame and the techniques used for discovering other devices within an Ethernet based network. Additionally, the simulation mode of the Cisco Packet Tracer tool is explored.

2. Theoretical considerations

This practical activity is concerned with Layer 2 operations performed on switches. Layer 2 refers to the Data Link Layer of the ISO/OSI model, which corresponds to the Network Access Layer in the TCP/IP model.

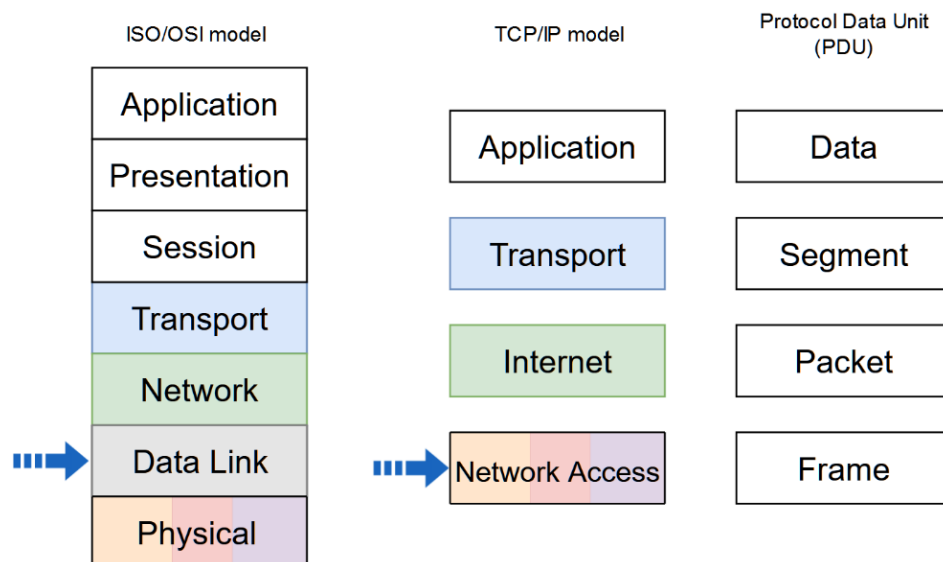


Figure 9.1 Network stack models and PDU naming in each level. The arrows indicate the addressed layers in the current activity

In order for a switch to forward a packet on a specific port, it maintains a switching table which contains a correspondence between a destination MAC address and the switch's port number. The MAC addresses used for communication are found in the Ethernet (Layer 2) frame header and a switch does not decapsulate the frame any further when manipulating the packet contents (Figure 9.2). This practical activity continues with providing more details regarding Layer 2 operations.

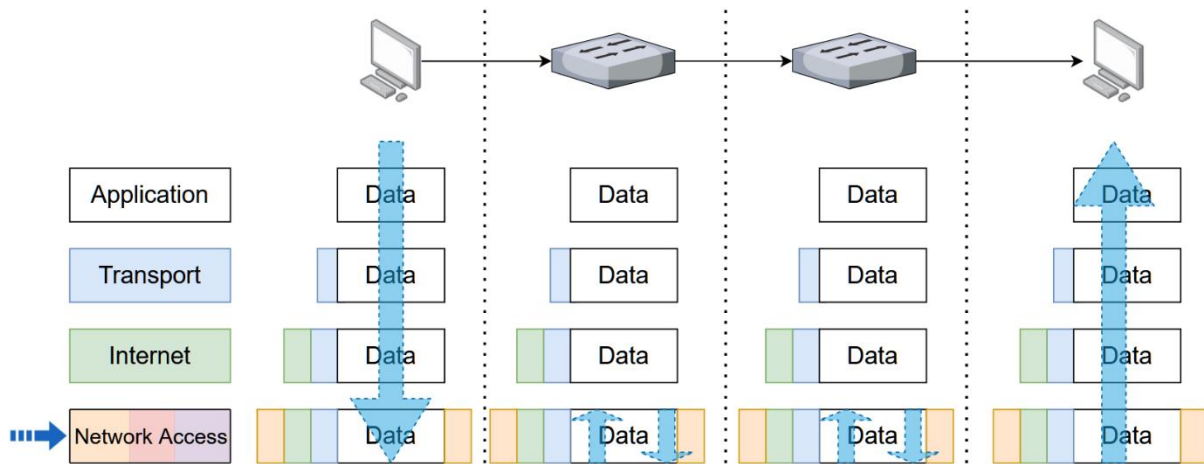


Figure 9.2 Switching operation, showing frame *serialization/deserialization* in the *Network Access* layer

2.1 Ethernet, Ethernet II and IEEE 802.3

Ethernet, Ethernet II and IEEE 802.3 are often used interchangeably as terms. Even though the terms tend to refer to very similar standards, they are slightly different, both historically and technically. In 1981 a consortium formed by Digital Equipment Corporation, Intel and Xerox (abbreviated as DIX) developed the Ethernet standard (also referred to as DIX 1.0 or Ethernet I). It was replaced with DIX 2.0, much more commonly known as Ethernet II in 1982. In 1983, IEEE introduced the 802.3 standard in an attempt to standardize the protocol beyond the DIX consortium. Nowadays, Ethernet II is generally the more popular approach, for reasons that will be described shortly.

There are two main differences between Ethernet II and IEEE 802.3. The first one is that Ethernet II uses a Type (also referred to as EtherType) field, which specifies the protocol encapsulated within the payload, whilst 802.3 uses that field for specifying payload length. The second difference is that, in order to run 802.3 within a TCP/IP stack, some additional information needs to be used (based on the SNAP and 802.2 format – beyond the scope of this practical activity) and, as such, taken from the Data field. This totals to 8 bytes which 802.3 uses from the Data field, reducing this field down to a range of 38 to 1492 bytes. Historically though, the length field has been considered not necessary and networks operate just as fine without it – this is the reason why Ethernet II is the more commonly used standard. All modern operating systems however work with both 802.3 and Ethernet II.

Note that when most engineers refer to Ethernet, they are generally referring to Ethernet II or, more rarely, to the IEEE 802.3 standard. This practical activity will use the term Ethernet and Ethernet II interchangeably, since Ethernet I is no longer used.

2.2 Ethernet II Frame Structure

Figure 9.3 presents the Ethernet II/IEEE 802.3 frame structure and the number of bytes allocated for each field. The following section describes the meaning of each field within an Ethernet frame.

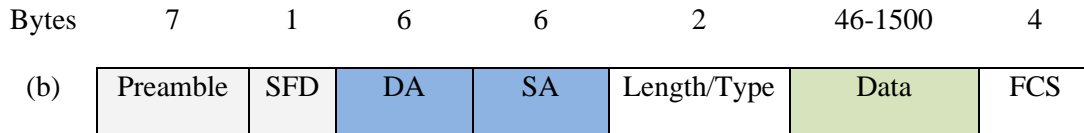


Figure 9.3 Ethernet II / IEEE 802.3 Frame Structure

Since Ethernet defines protocols for both the Physical and the Data Link Layer parts of a networking stack, some fields are handled by the Physical layer (Preamble and SFD), whilst some by the Data Link Layer (other fields).

The Preamble field is a 56-bit series of alternating ‘0’ and ‘1’ bits. They are used so that the devices involved in communication can synchronize their respective clocks and thus adjust the sampling rate accordingly for correct reception of the frame. The concept of using a preamble does not impose a fixed length but is rather adjusted on an individual protocol basis, even if Ethernet uses a fixed length of 56 bits. Using more bits allows more time for the communicating devices to synchronize but increases the overhead of communications, whilst reducing the preamble length has opposite effects.

The Start Frame Delimiter (SFD) field is a byte used to break the bit pattern in the Preamble and mark the start of the rest of the Ethernet frame. Specifically, it is “10101011” or 0xD5 (again, this is specifically in Ethernet; other protocols might use different SFD values). Please consider the fact that the bits are transmitted left to right and interpreted in LSB order.

The Destination Media Access Control (MAC) Address (DA) and Source MAC Address (SA) are identifiers which are uniquely assigned to the Network Interface Controller (NIC) of each device. Note that a device can have multiple NICs and, therefore, multiple corresponding MAC addresses. The role of the DA and SA will be discussed in more detail in the following sections.

The Type field is used to indicate the type of message encapsulated in the frame. Table 9.1 indicates some specific Type values.

Table 9.1 EtherType Examples

Hex Value	Protocol Type
0x0000-0x05DC	Length field for IEEE 802.3
0x0600	Xerox
0x0800	IPv4
0x0801	X.75
0x0806	ARP
0x86DD	IPv6

Note that due to the minimum requirement of 46 bytes used for data transmission if the length is any less than that value the Data Link Layer adds padding bytes to the Data field. The 46 byte value is based on the CSMA/CD mechanism (presented during the lecture) and beyond the scope of this activity. Alternatively, this field is considered to represent Length for 802.3, when its value is less than 0x05DC.

The Data field corresponds to the payload which is encapsulated within the frame. This is typically higher level protocol data.

The Frame Check Sequence (FCS) field is used for verifying integrity of the message. It is a four-byte Cyclic Redundancy Check (CRC). It is a numerical value which is computed based on all data within the frame, with the exception of the actual FCS (and, obviously, the Preamble and SFD). On reception this value is recalculated and compared with the original FCS. If the two values are different then the frame contains errors and is discarded.

2.3 Address Resolution Protocol

The Address Resolution Protocol (ARP) is a very important protocol in networking. As seen during the lectures and previous activities, addressing is handled separately by OSI (or TCP/IP) stack layers. The DLL handles MAC addressing (even though they are sometimes referred to as physical addresses and are dependent on the NIC, the Physical layer does not generally handle MAC addresses), the network layer handles IP addresses and the Transport layer handles port numbers. The Transport layer is not addressed in this practical activity. In a typical networking scenario, when a device intends to send a message to a destination, it already knows the destination IP address from a DNS server. However, in order to correctly assemble a frame, the DLL needs to know the MAC address, which is not handled by DNS servers and manual handling is extremely impractical. As such, ARP provides a simple mechanism to figure out the MAC address for a known IP address, a process known as *Address Resolution*.

Each device contains an internal data structure, known as an ARP cache, which stores the mappings between IP addresses and MAC addresses on a network. ARP is used to populate this cache. Figure 9.4 illustrates the contents of the cache through running the Windows *arp -a* console command.

```
C:\Users\admin>arp -a

Interface: 192.168.0.103 --- 0x12
 Internet Address      Physical Address      Type
 192.168.0.1          c4-6e-1f-37-70-61    dynamic
 192.168.0.101       9c-2e-a1-ed-55-ab    dynamic
 192.168.0.255       ff-ff-ff-ff-ff-ff    static
```

Figure 9.4 ARP Cache

In order to do so two ARP frames are generally needed: an *ARP Request* and an *ARP Reply* frame. Let us consider two devices on a network: device A intends to transmit a message to device B. The ARP algorithm is as follows:

1. Device A checks its ARP cache. If there is an entry with B's IP address it will jump to step 5
2. Device A broadcasts an ARP request containing the target IP. All devices receive this broadcast since A doesn't yet know the MAC address of B
3. If device B is on the network it will reply with an ARP response containing its own MAC address. All other devices will silently (i.e. without sending a message announcing this) discard the request
4. Device A will update its ARP cache
5. Device A will send the intended message as a unicast to B

ARP caches contain two types of entries: static and dynamic. Static entries are introduced by the user and are kept permanently in the cache, unless specifically removed. Dynamic entries are introduced by ARP and are periodically deleted. Each ARP entry can be deleted after periods of seconds, up to several hours, depending on the network, the device type, OS features and individual configurations. Deleting dynamic entries is an automatic process (but which can be initiated manually) and is useful because some ARP entries might not be needed anymore. Some examples of this situation are:

- A device changes its IP address (especially if using DHCP)
- A device is removed from the network so the entry might not be needed anymore
- A device has its NIC changed and, implicitly, its corresponding MAC address in the network

The reason for periodically removing entries in the cache is to remove any unnecessary or unused entries (especially since the cache is of fixed size, which might lead to certain strategies of cyber-attacks). Only ARP messages update ARP caches, so a device will update an entry either when receiving an ARP request, when receiving an ARP reply as part of the resolution process or when receiving an ARP broadcast (this final scenario only updates ARP entries, it does not add new ones – also used for avoiding overflowing the cache).

Static entries should generally be used only when a device is intended to remain in the network for a long time (e.g. a router). With the exception of some forms of cyber-attacks, ARP does not generate much overhead.

Other ARP use cases are: using Proxy ARP, which implies one device responding to an ARP request on behalf of another device and using a gratuitous ARP, which implies sending an ARP broadcast so that other hosts can update their respective entries. These use cases are beyond the scope of this activity.

2.4 Neighbor Discovery Protocol

Neighbor Discovery Protocol (NDP or simply ND) is a protocol used with IPv6 which has multiple roles. It defines five ICMPv6 packet types, some of which have already been presented. These are: Router Solicitation (RS), Router Advertisement (RA), Neighbor Solicitation (NS), Neighbor Advertisement (NA) and Redirect packets.

NDP fulfills several roles, of which the current activity work only briefly presents MAC and IPv6 address resolution.

In IPv6 NS and NA messages are used to replace ARP, and are, to a certain extent, equivalent to the ARP Request and ARP Reply messages. Much like the ARP cache, IPv6 enabled devices use an IPv6 Neighbor Table or IPv6 Neighbor Discovery Cache. There are, however, certain optimizations with NDP.

One significant optimization is brought on by the use of multicast addresses: instead of sending a broadcast ARP request, NDP implies sending an NS to the target device's multicast address, which reduces network overhead.

Another optimization is brought on by using five states which describe an IPV6 ND Cache entry:

1. Incomplete (NS has been sent and NA not yet received)
2. Reachable (NS has been sent and NA received or ND entry successfully used by upper layer protocol)
3. Stale (Timeout interval elapsed)
4. Delay (Timeout interval elapsed but recent packet sent to target, state moving to Probe after sending an NS)
5. Probe (NS has been sent from delay, waiting for NA)

IPv6 ND builds upon ARP but has multiple functions and is much more complex than simply resolving MAC addresses to IP addresses.

3. Practical activity

In the following activity you will use Wireshark in order to analyze the ARP protocol. Using the local ARP cache on the device requires administrator privileges. The current instructions are for Windows based systems. On Unix based systems, the **sudo** command might be required to manipulate the local ARP cache. The activity has two parts:

1. Working on the local device
 - a. Clearing the local ARP cache
 - b. Examining an ARP request
 - c. Examining an ARP reply

2. Working in Packet Tracer
 - a. ARP Simulation Mode
 - b. NDP Simulation Mode

3.1 Working on the local device

Step 1: First you will need to open a command line or PowerShell with Administrator privileges. To do this, right click the appropriate program and select Run as Administrator. Enter the password when prompted. Use the **ipconfig /all** command and write down your IPv4 address, your appropriate MAC address and the default gateway's IPv4 address.

Step 2: Open a Wireshark capture on the appropriate interface. In order to clear the ARP cache you need to use the **arp -d** command. To view the ARP cache the **arp -a** command is used. Since the ARP cache is continuously updated, in order to make sure that it is cleared you can combine the two instructions using the **&** character, as follows: **arp -d & arp -a** (optional, if **arp -d** does not work, use one of the following commands: **arp -ad** or **netsh interface ip delete arpcache**). To make sure that the MAC address of the default gateway is reintroduced in the cache ping the default gateway's IPv4 address. Stop the Wireshark capture. You can use the **arp -a** command again to check that the cache now contains the default gateway's corresponding entry.

Step 3: Use the **arp** filter in Wireshark in order to view only ARP frames. Select the first broadcast message. This is an ARP request message. Notice that the type is Ethernet II (unless specifically using a different protocol). Expand the appropriate Ethernet II tab in Wireshark. Check that the message originated from your device either using the source MAC address or the source IP address (there might be other request messages on the network). If the first request isn't yours, go ahead and navigate until you find your own.

Step 4: Now that you've identified the correct ARP request message go ahead and analyze it (Figure 9.5). Notice that the DA field is **FF:FF:FF:FF:FF:FF**. This is a broadcast address. Observe that the type field is **0x0806** which correctly indicates an ARP frame. Note that the addresses in your own case will be different.

```

Ethernet II, Src: IntelCor_14:1b:5d (7c:5c:f8:14:1b:5d), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
  Destination: Broadcast (ff:ff:ff:ff:ff:ff)
    Address: Broadcast (ff:ff:ff:ff:ff:ff)
      .... ..1. .... .. = LG bit: Locally administered address (this is NOT the factory default)
      .... ..1. .... .. = IG bit: Group address (multicast/broadcast)
  Source: IntelCor_14:1b:5d (7c:5c:f8:14:1b:5d)
    Address: IntelCor_14:1b:5d (7c:5c:f8:14:1b:5d)
      .... ..0. .... .. = LG bit: Globally unique address (factory default)
      .... ..0. .... .. = IG bit: Individual address (unicast)
  Type: ARP (0x0806)

```

Figure 9.5 Wireshark capture of a detailed ARP Request frame

Step 5: Let us investigate the actual ARP contents. Expand the appropriate selection, as seen in Figure 9.6.

```

Address Resolution Protocol (request)
  Hardware type: Ethernet (1)
  Protocol type: IPv4 (0x0800)
  Hardware size: 6
  Protocol size: 4
  Opcode: request (1)
  Sender MAC address: IntelCor_14:1b:5d (7c:5c:f8:14:1b:5d)
  Sender IP address: 192.168.100.5
  Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
  Target IP address: 192.168.100.1

```

Figure 9.6 Wireshark capture of an ARP Request frame

Let us analyze each field and understand their respective purpose and meaning. Hardware type and Protocol type refer to what types of addresses are being mapped to one another. In this case a MAC address is mapped to a known IPv4 address (remember that this is the purpose of ARP). The following two fields refer to the size of each address: a MAC address is 6 bytes long whilst an IPv4 address is 4 bytes long. The Opcode is, in the case of ARP, one of two options: “1” represents a request and “2” represents a response. The Sender MAC and IP addresses are obviously your own (including the sender MAC in the request ensures that the reply can be sent as unicast to the requester). What is noteworthy though is that the protocol includes the Target MAC and Target IP addresses. A **very important** distinction is the use of *Target* instead of *Destination*. Even though the destination is a broadcast, as previously seen, the target represents the device whose MAC address is being resolved. Hence the distinction between destination and target. The IP address is clearly the default gateway and because the target MAC address has yet to be resolved this field is left unpopulated.

Step 6: Let us have a look at the corresponding ARP reply (Figure 9.7). First you need to find the correct reply – it should be from the default gateway to your device. Depending on how long the capture was running for there might be multiple requests and replies – this is due to the ARP cache refresh rate.

```

Ethernet II, Src: HuaweiTe_e5:99:36 (04:fe:8d:e5:99:36), Dst: IntelCor_14:1b:5d (7c:5c:f8:14:1b:5d)
  Destination: IntelCor_14:1b:5d (7c:5c:f8:14:1b:5d)
    Address: IntelCor_14:1b:5d (7c:5c:f8:14:1b:5d)
      .... ..0. .... = LG bit: Globally unique address (factory default)
      .... ..0. .... = IG bit: Individual address (unicast)
  Source: HuaweiTe_e5:99:36 (04:fe:8d:e5:99:36)
    Address: HuaweiTe_e5:99:36 (04:fe:8d:e5:99:36)
      .... ..0. .... = LG bit: Globally unique address (factory default)
      .... ..0. .... = IG bit: Individual address (unicast)
Type: ARP (0x0806)

```

Figure 9.7 Wireshark capture of an ARP Reply frame

It can be seen that this is a unicast message from a MAC on the network (check that it’s the default gateway based on what you previously noted).

Step 7: Let us investigate the contents of the actual protocol (Figure 9.8).

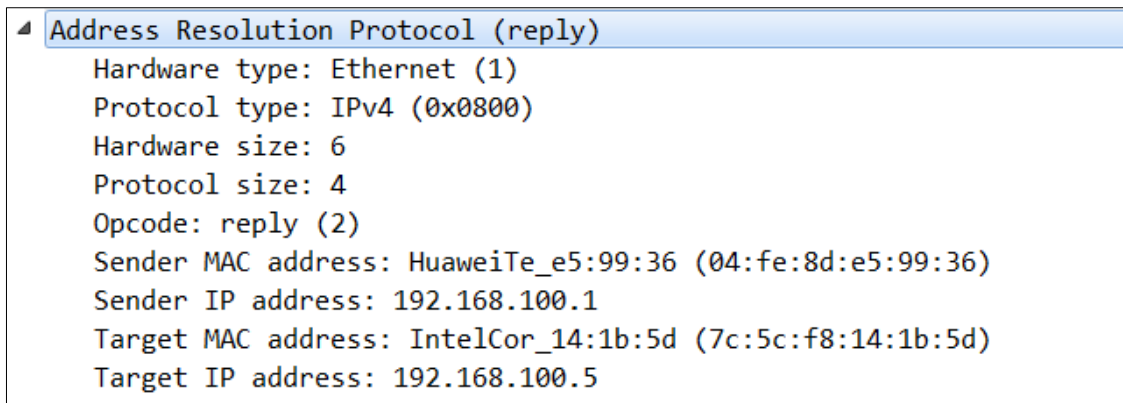


Figure 9.8 Wireshark capture of an ARP Reply frame

Notice the Opcode is changed and that the Sender MAC address is now visible (during the request this corresponded to the unknown Target MAC address). In conclusion, our own device receives this reply from the default gateway and can thus populate its ARP cache with the appropriate MAC address. Communications can continue now without exchanging any more ARP messages, except if the entry is deleted after a timeout.

3.2 Working in Packet Tracer

a. ARP Simulation Mode

This part of the practical activity uses the Simulation mode of the Cisco Packet Tracer tool to verify how network packets travel inside a network. This will also clarify why the first echo request of a **ping** command can sometimes be an unsuccessful timeout (as probably noticed in previous activities).

Step 1: Launch Packet Tracer, create a network topology containing only switches and endpoint devices and navigate to the Simulation mode, as indicated by the arrow in Figure 9.9.

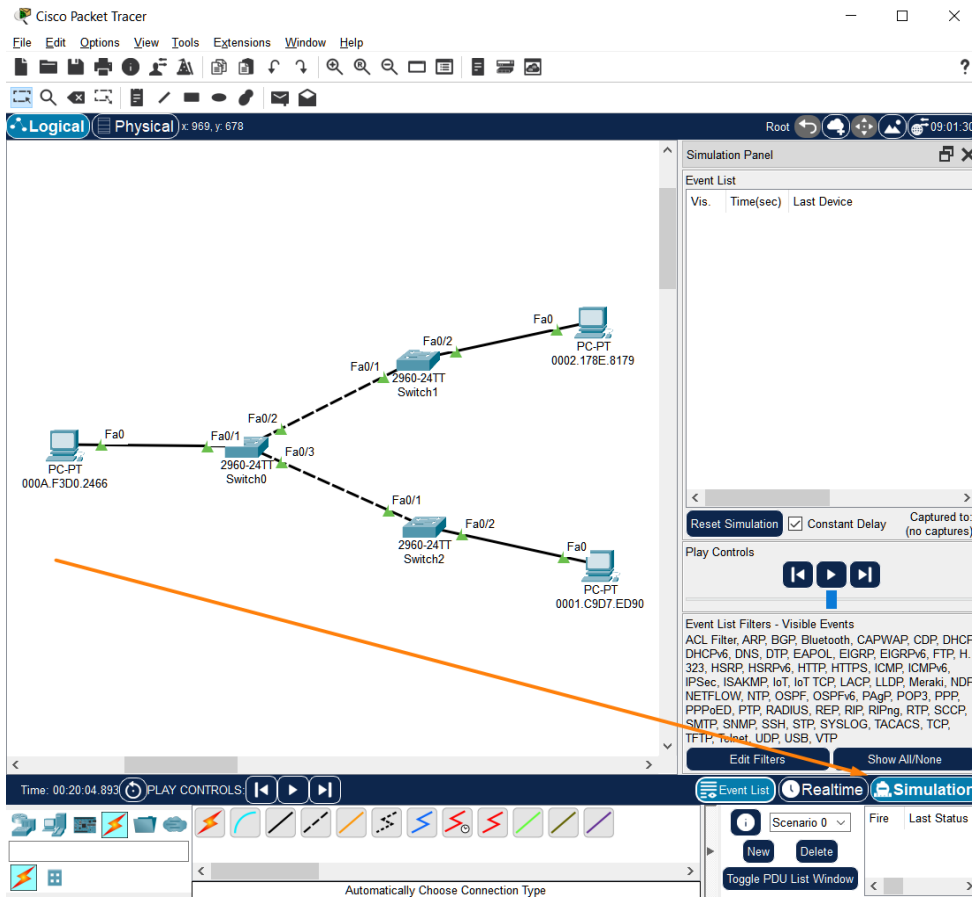


Figure 9.9 PacketTracer Simulation Mode

Step 2: In the Simulation window, click on the Show All/None button to clear all filters and then click on the Edit Filters button and select only ARP and ICMP (Figure 9.10).

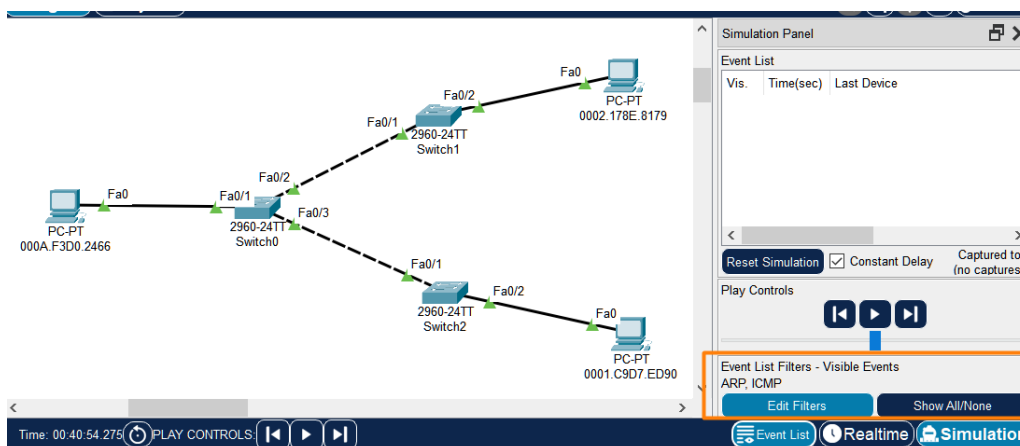


Figure 9.10 PacketTracer Simulation Filters

Step 2: Rename the PC names with their own MAC addresses; they can be found in the PC menu -> Config tab -> FastEthernet0 interface .

Step 3: Assign each PC an IP address from the same network (e.g. 10.0.0.1, 10.0.0.2 and 10.0.0.3, all of them /8 – feel free to use a different network/subnet).

Step 4: Open the command prompt on one of the PCs and verify that its arp cache is empty. If it is not empty run the **arp -d** command to clear it.

Step 5: Ping another PC's IP address and inspect the simulation. At this point the ARP cache of the PC is empty so it cannot populate the entire packet (specifically it cannot populate the Ethernet frame DA field), therefore it launches an ARP broadcast request in the network – recall that this is the first stage of the ARP protocol. The step by step traffic analysis (Figure 9.11, Figure 9.12 and Figure 9.13) shows the path that the request takes; note that only the target device replies to the broadcasted request and all other devices silently drop the packet.

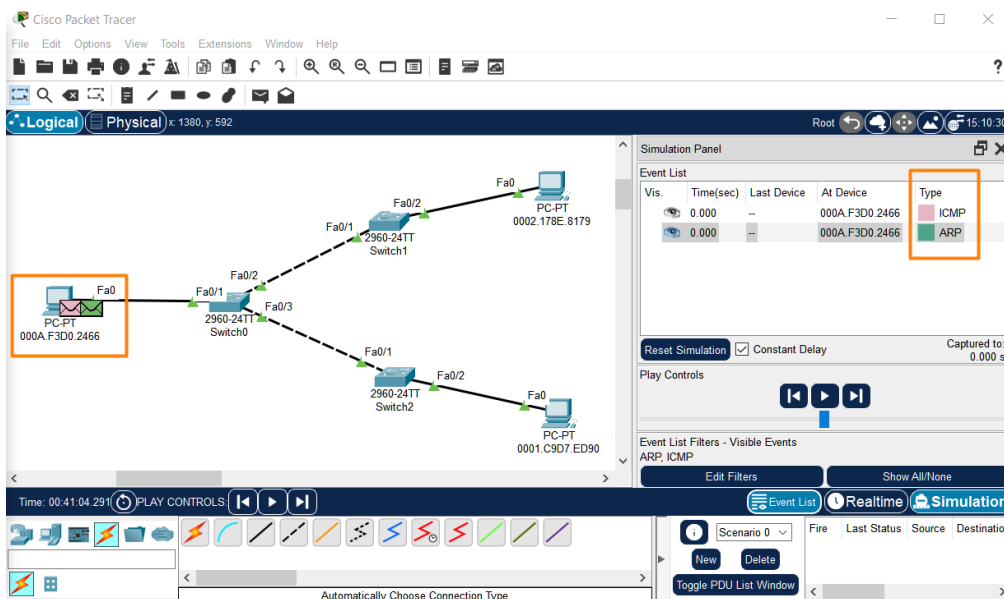


Figure 9.11 ARP request

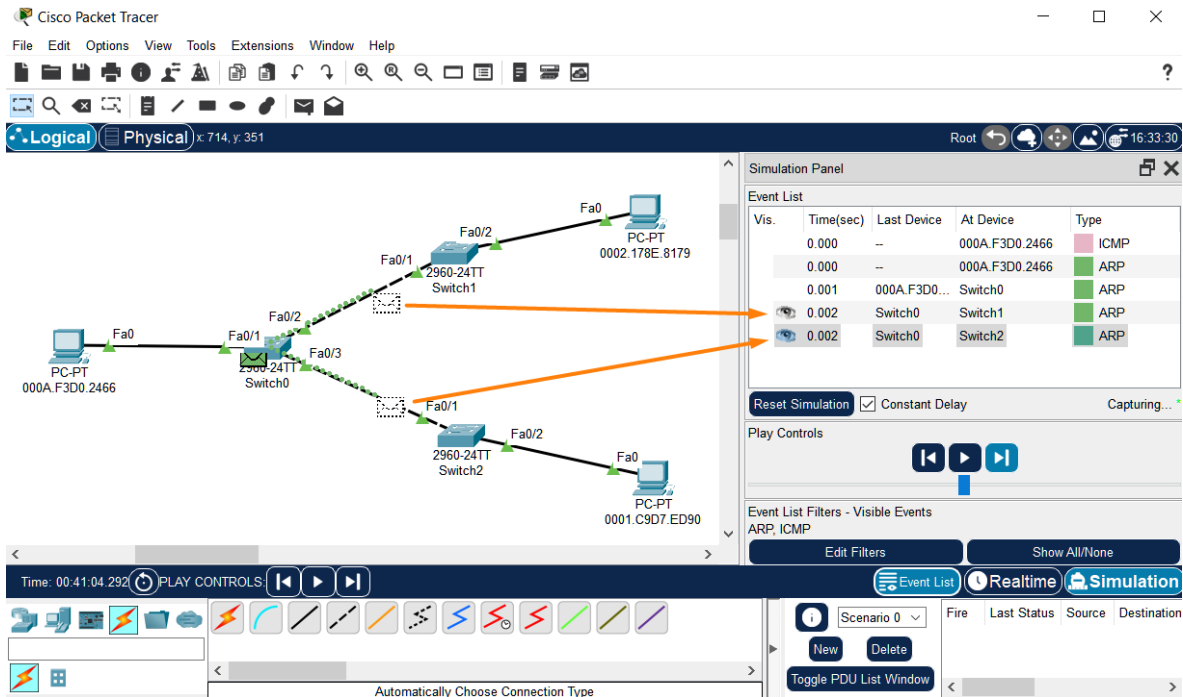


Figure 9.12 ARP broadcast

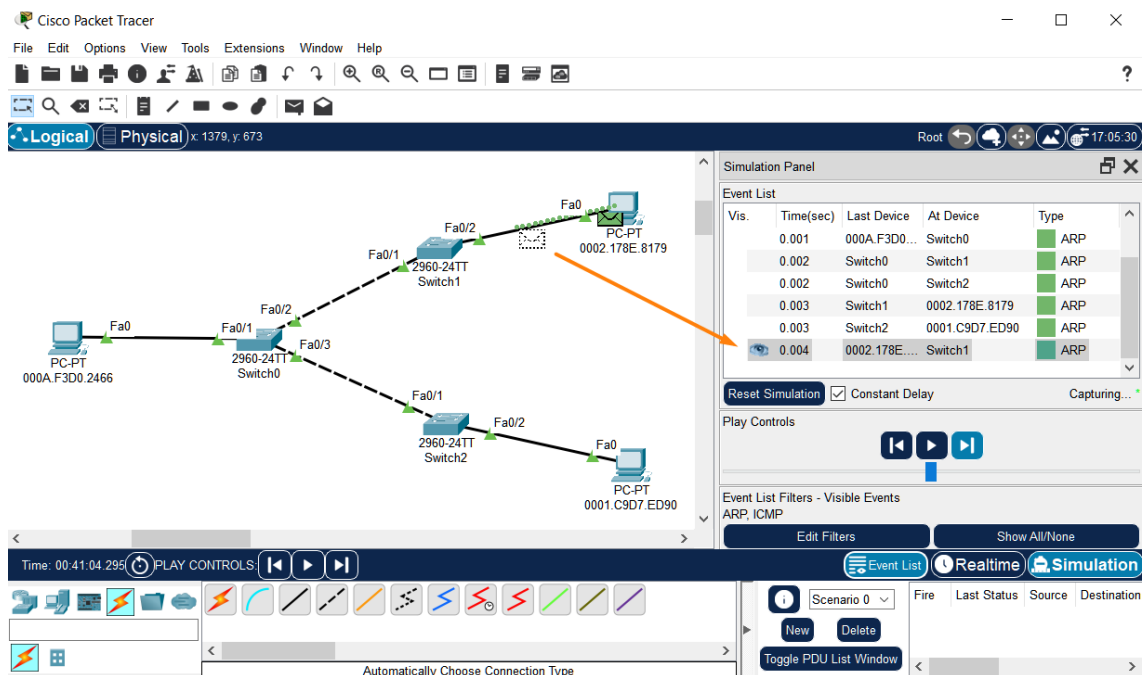


Figure 9.13 ARP reply

Step 6: The packet content can be explored by double clicking the packet in the Event List (Figure 9.14). The same information which was discovered in the first part of the practical activity can be seen here (information at all 7 layers of the OSI model, the same as in Wireshark).

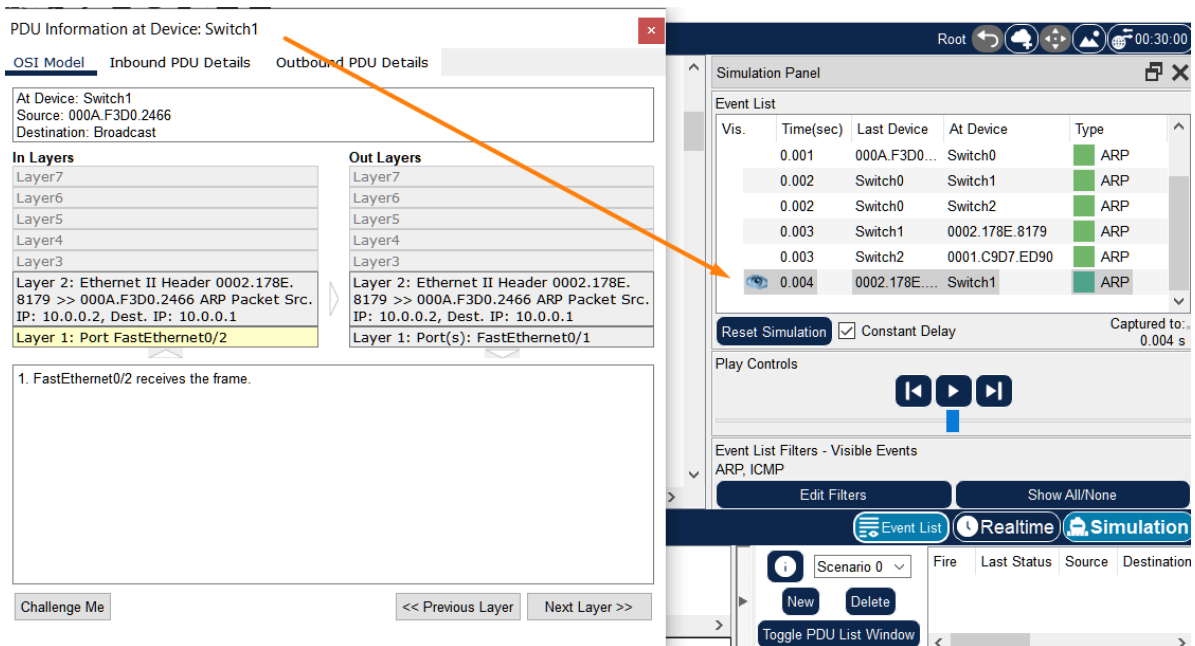


Figure 9.14 Inspection of frame content in PacketTracer

Step 7: Resume the packet flow in the network and inspect the console of the PC running the **ping** command. If the ARP reply takes too long to return then the first ICMP echo reply message might not reach the PC in time, resulting in a request timeout (recall that the **ping** utility uses ICMP echo requests and replies). This explains why you probably noticed in previous Packet Tracer activities that, especially on a newly formed network, some messages time out (Figure 9.15).

```
C:\>ping 10.0.0.2

Pinging 10.0.0.2 with 32 bytes:

Request timed out.
Request timed out.
Reply from 10.0.0.2: bytes=32
```

Figure 9.15 Ping timeout exemplified in PacketTracer

Step 8: Use the following commands on the switch components to inspect their MAC address tables and how they get populated when the first ARP requests/replies travel through the network.

Switch>enable

Switch#show mac address-table

Switch#clear mac address-table

b. NDP Simulation Mode

Using the IPv6 .pkt file which was created in a previous activity for the static routing functionality, apply the NDP packet filter in the Edit Filters window and inspect the traffic according to the description in the activity text. Find the RS, RA, NS and ND packets using the Simulation mode.

CHAPTER 10: VLANs, TRUNKING AND INTER-VLAN ROUTING

1. Objectives

At the end of the practical activity, students will be able to define and classify Virtual Local Area Networks (VLANs), explain the purpose of trunking and inter-VLAN routing and configure VLAN-based networks in a multi-switched environment.

2. Theoretical considerations

The current practical work focuses on the Data Link and Network layers of the ISO/OSI stack (Figure 10.1).

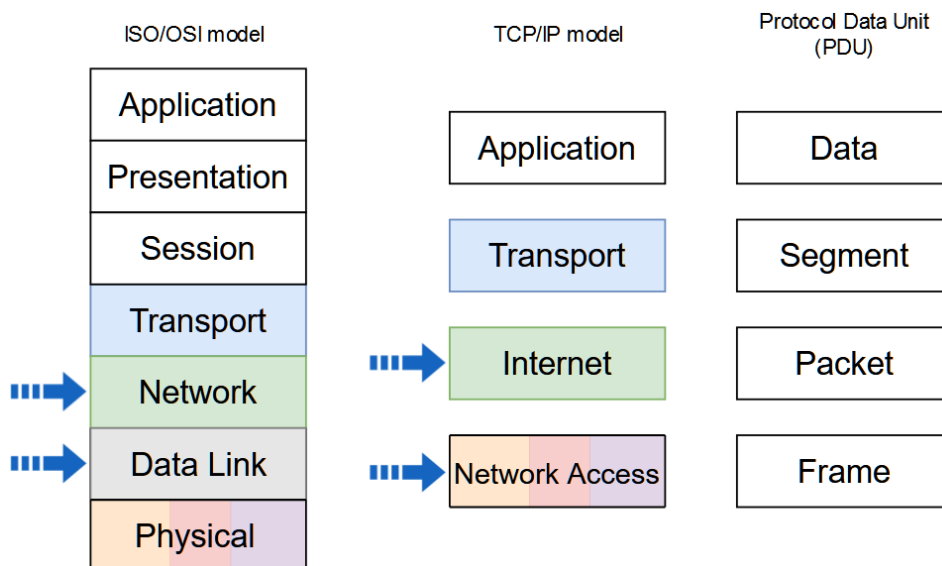


Figure 10.1 Network stack models and PDU naming in each level. The arrows indicate the addressed layers in the current activity

2.1 VLANs

A VLAN is a partition of the set of devices connected to the local network. Grouping into VLANs can be done according to different criteria such as the role of users or the type of traffic. This grouping can be done regardless of the physical location of the devices or users (Figure 10.2). VLANs work by logically segmenting the network into broadcast domains, with each VLAN representing a different broadcast domain. The switch maintains a different bridging table for each VLAN. Devices in a VLAN are restricted to communicating only with devices in the same VLAN. Connectivity between VLANs is facilitated by routers.

The benefits of VLANs are:

- smaller broadcast domains;
- reduced cost;
- increased network performance;
- increased scalability;
- increased security;
- better management.

Common types of VLANs:

- Default VLAN – Also known as VLAN 1, cannot be deleted or renamed. All switch ports are members of VLAN 1 by default;
- Data VLAN – Data VLANs are commonly created for specific groups of users or devices. They carry user generated traffic;
- Voice VLAN – Voice VLAN is created because this type of traffic requires assured bandwidth and delay less than 150 ms from source to destination;
- Native VLAN – This is the VLAN that carries all untagged traffic. This is traffic that does not originate from a VLAN port;
- Management VLAN – This is a VLAN that is created to carry network management traffic including SSH, SNMP, Syslog, and more.

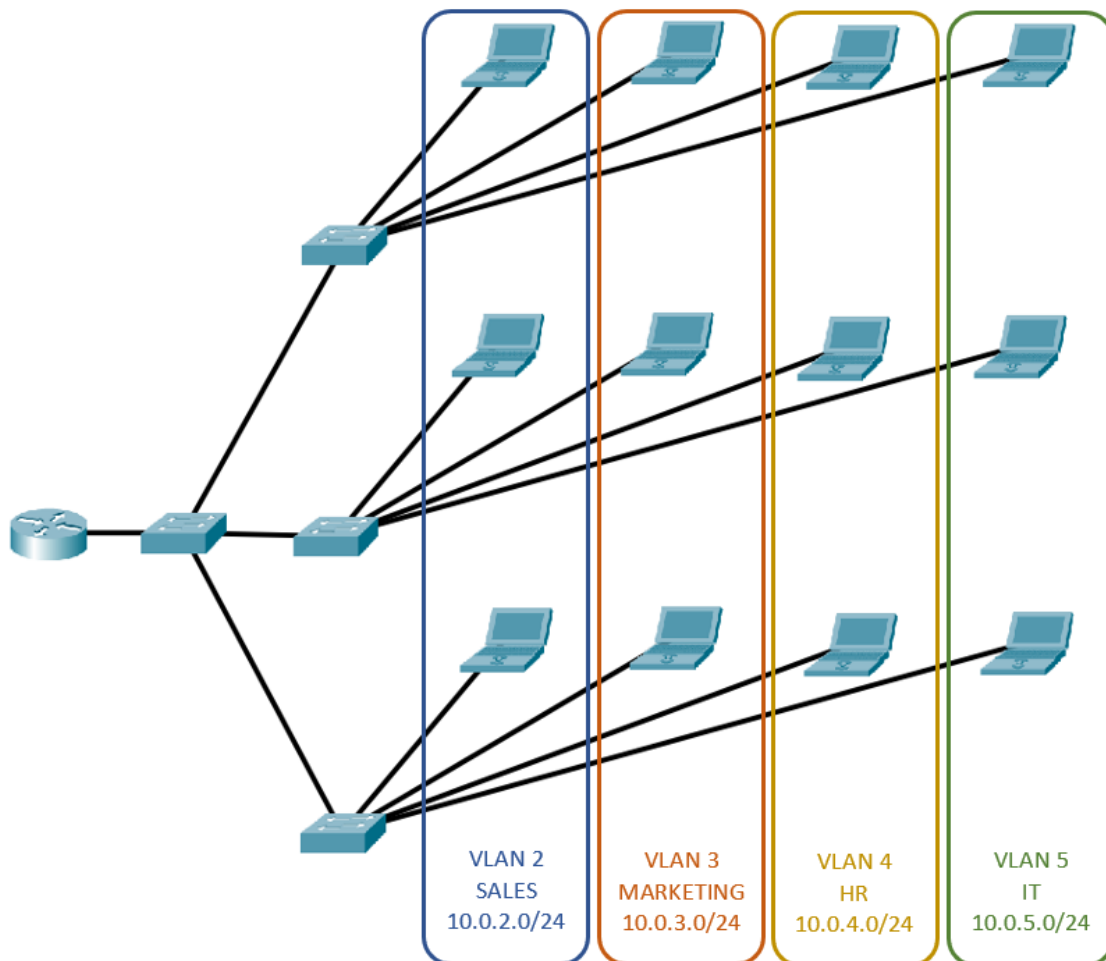


Figure 10.2 VLANs in a multi-switched environment

2.2 Trunking

A trunk is a point-to-point link between two network devices that does not belong to a specific VLAN and carries more than one VLAN. It extends VLANs across the network and enables devices connected to different switches, but in the same VLAN, to communicate through the switched network.

The ports assigned to VLANs are configured in access mode and use standard Ethernet frame headers. This header does not contain information about the VLAN to which the frame belongs. When the frames are forwarded between switches on trunk lines, the VLAN membership information must be transmitted with the frames. Therefore, when Ethernet frames are placed on the trunk, the VLAN membership information is added, the frames using 802.1Q headers instead of Ethernet headers. Adding information about VLANs is called tagging, and 802.1Q headers also add other information to the frames beside VLAN membership.

The Figure 10.3 presents the Ethernet II/IEEE 802.3 frame structure used in ports configured in access mode and the IEEE 802.1Q frame structure used in ports configured in trunk mode. The following section describes the meaning of the Tag control information fields.

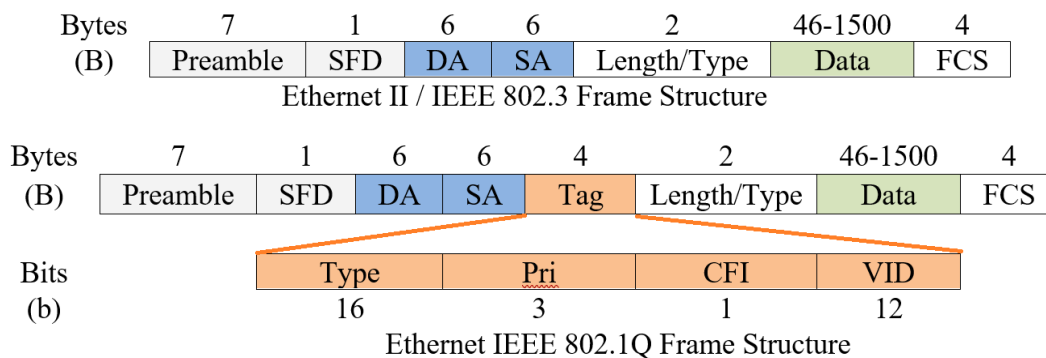


Figure 10.3 Ethernet II/IEEE 802.3 and IEEE 802.1Q frames

VLAN tag control information field consists of the following fields:

- Type - Tag protocol ID (TPID) value. For Ethernet, it is set to hexadecimal 0x8100.
- User priority - Supports level or service implementation.
- Canonical Format Identifier (CFI) - Enables Token Ring frames to be carried across Ethernet links.
- VLAN ID (VID) - VLAN identification number, supports up to 4096 VLAN IDs.

In the example below (Figure 10.4), Laptop1 connected to switch S2 on access port Fa0/6 in VLAN 10 is communicating with Laptop2 connected to another switch, S3, on access port Fa0/7 in the same VLAN, VLAN 10. The ports between the switches are configured in trunk mode. Laptop 1 sends a packet to Laptop 2. When the packet enters switch S2 on access port Fa0/6, the packet is encapsulated into an Ethernet II/IEEE 802.3 frame. The S2 switch forwards the packet on the Fa0/1 trunk port encapsulating the packet into an Ethernet 802.1Q frame. The VLAN number is set to 0x00a (VLAN 10).

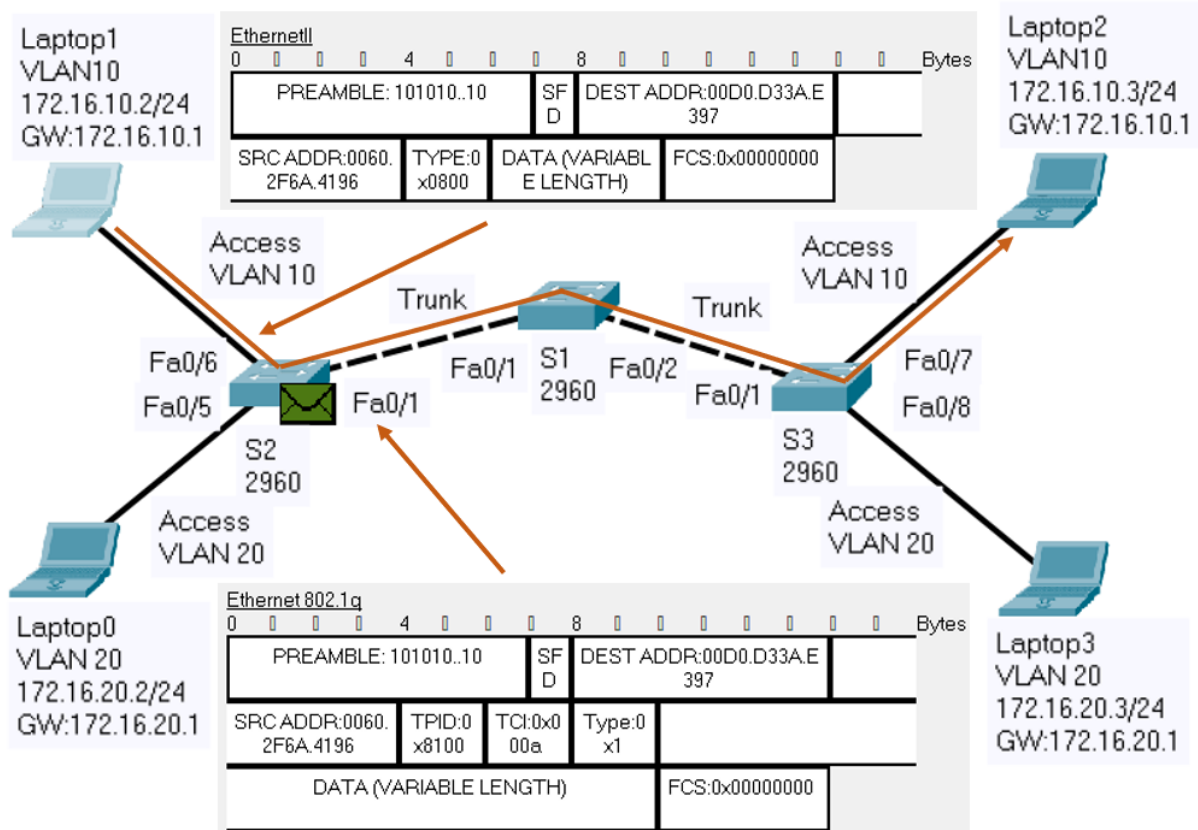


Figure 10.4 Communication in the same VLAN

2.3 Inter-VLAN routing

Layer 2 switches don't forward traffic from one VLAN to another. The traffic between VLANs is forwarded using Layer 3 devices, routers or Layer 3 switches, the process being called Inter-VLAN routing. There are three options for inter-VLAN routing:

- Legacy inter-VLAN routing;
- Router-on-a-Stick;
- Layer 3 switching using SVIs.

The router-on-a-stick approach (see Figure 10.5) uses one of the router's physical interfaces for inter-VLAN routing.

- Logical subinterfaces are created on the physical interface; one subinterface per VLAN; the subinterfaces use 802.1Q encapsulation to process VLAN tags;
- Each VLAN is assigned a different network/subnetwork address;
- Each subinterface is configured in a VLAN with an IP address from the VLAN it represents;
- VLAN hosts are assigned IP addresses from their corresponding VLANs; each host is configured to use as default gateway the subinterface representing its VLAN.
- When a host in a VLAN communicates with a host in a different VLAN, it sends the packets to its own gateway, in its own VLAN; the router internally routes between the VLANs using subinterfaces as the VLAN networks are present in the routing table as connected; the router receives the packets on the source VLAN subinterface and

forwards the routed traffic as VLAN-tagged for the destination VLAN out the trunk link

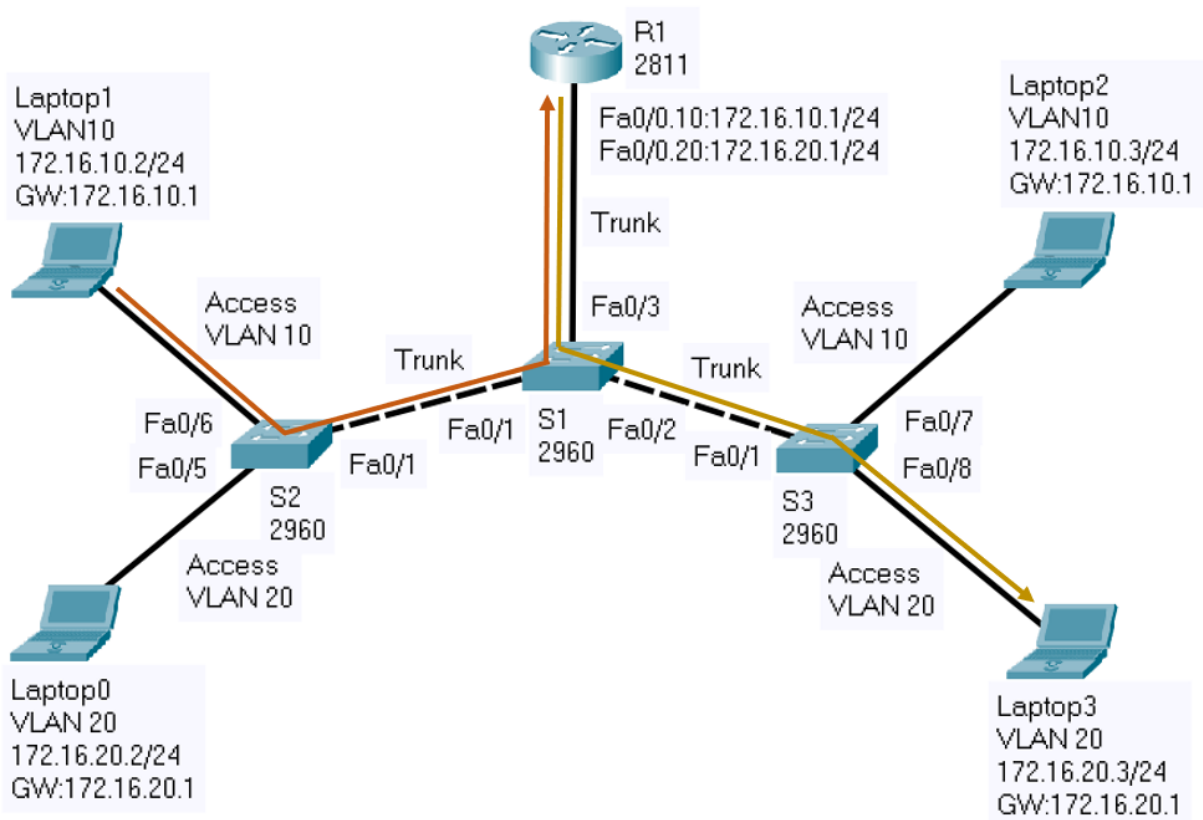


Figure 10.5 Router-on-a-stick option for inter-VLAN routing

3. Practical activity

3.1 Discuss the theoretical aspects presented in this chapter.

3.2 Consider the network topology in Figure 10.6:

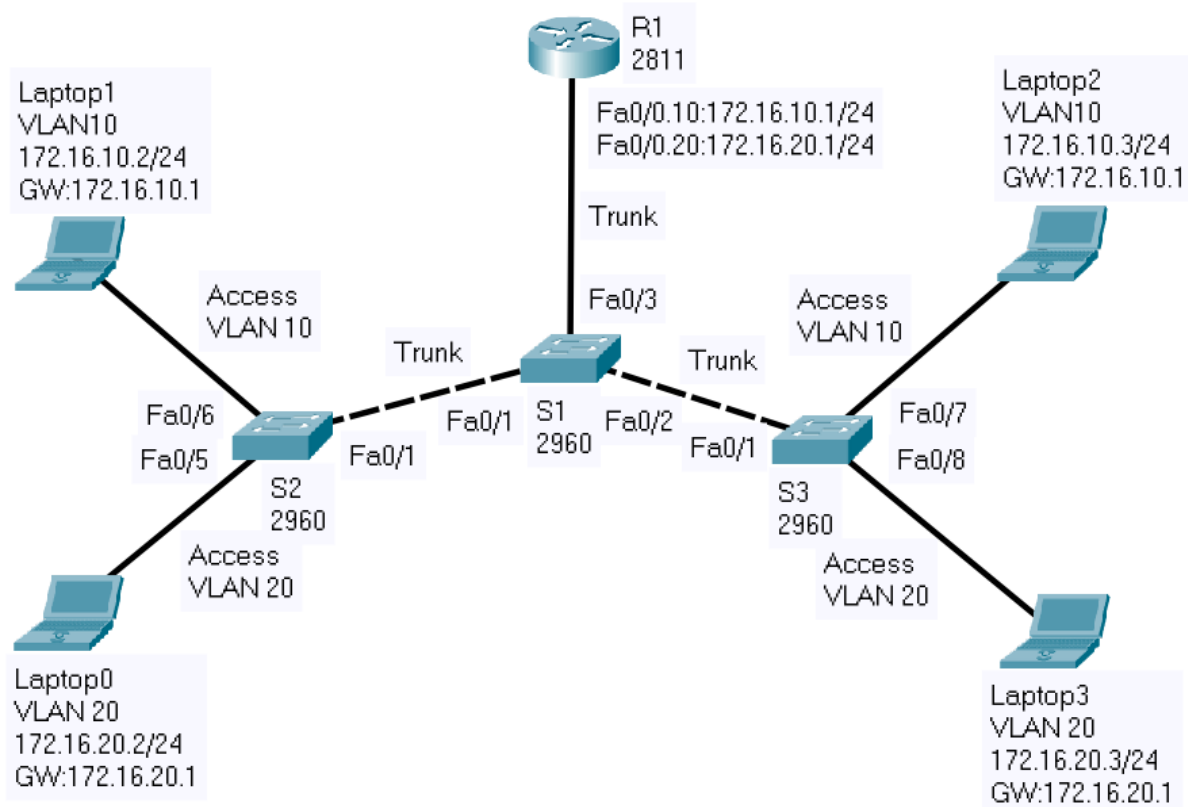


Figure 10.6 Test network topology

Step 1: Before configuring the network devices, discuss the IPv4 address assignment in the Table 10.1:

Table 10.1 IPv4 addresses for the test network

Device	Interface	IP Address	Netmask	Gateway
Laptop 0	Fa0	172.16.20.2	255.255.255.0	172.16.20.1
Laptop 1	Fa0	172.16.10.2	255.255.255.0	172.16.10.1
Laptop 2	Fa0	172.16.10.3	255.255.255.0	172.16.10.1
Laptop 3	Fa0	172.16.20.3	255.255.255.0	172.16.20.1
R1	Fa0/0.10	172.16.10.1	255.255.255.0	-
R1	Fa0/0.20	172.16.20.1	255.255.255.0	-

Step 2: Specify the host names for the networking devices (router and switches)

General syntax:

Switch(config)#hostname host-name

Description: Specifies or modifies the host name

Example:

Switch(config)#hostname S2

Step 3: Create VLAN 10 and 20 on all the switches and verify vlan information

General syntax:

Switch(config)#vlan vlan_id

Description: Global configuration command that creates VLAN vlan_id

Switch(config-vlan)#name vlan_name

Description: Assigns a name to the VLAN

Example:

S2(config)#vlan 10

S2(config-vlan)#name Vlan10

S2(config-vlan)#exit

S2(config)#vlan 20

S2(config-vlan)#name Vlan20

General syntax:

Switch#show vlan

Switch#show vlan brief

Description: Displays VLANs information (the contents of the vlan.dat file)

Step 4: Assign ports to VLANs and verify the configuration

General syntax:

Switch(config)#interface interface_id

Description: Enters interface configuration mode

Switch(config-if)#switchport mode access

Description: Sets the port to access mode

Switch(config-if)#switchport access vlan vlan_id

Description: Assigns the port to a VLAN

Example:

```
S2(config)#interface fastEthernet 0/6
S2(config-if)#switchport mode access
S2(config-if)#switchport access vlan 10
S2(config-if)#exit
S2(config)#interface fastEthernet 0/5
S2(config-if)#switchport mode access
S2(config-if)#switchport access vlan 20
```

General syntax:

```
Switch#show vlan
Switch#show vlan brief
```

Description: Displays VLANs information (the contents of the vlan.dat file)

Step 5: Set the switch ports connected to other networking devices to trunk mode and verify the configuration

General syntax:

```
Switch(config)#interface interface_id
```

Description: Enters interface configuration mode

```
Switch(config-if)#switchport mode trunk
```

Description: Forces the link to be a trunk link

Example:

```
S2(config)#interface fastEthernet 0/1
S2(config-if)#switchport mode trunk
```

General syntax:

```
Switch#show interfaces trunk
```

Description: Displays trunking information for the switch

Step 6: Configure the hosts with the IP addressing information in the figure (IP address, netmask and gateway) and test the connectivity between them

- a. ping <target IP>
- b. tracert <target IP>

Step 7: Configure Inter-VLAN routing and test the connectivity between hosts in different VLANs

General syntax:

Router(config)#interface interface_id

Description: Enters interface configuration mode

Router(config-if)#no shutdown

Description: Enables the interface

Router(config-if)#exit

Description: Returns to the global configuration mode

Router(config)#interface interface_id.subinterface_id

Description: Creates a subinterface on an interface

Router(config-subif)#encapsulation dot1Q vlan_id

Description: Specifies IEEE 802.1Q as the VLAN tagging method for VLAN *vlan_id* on this subinterface

Router(config-subif)#ip address ip_address netmask

Description: Adds an IP address and a netmask on this subinterface

Router(config-subif)#end

Description: Returns to the privileged exec mode

Router#show ip route

Description: Displays the routing table

Example:

R1(config)#interface fastEthernet 0/0

R1(config-if)#no shutdown

R1(config-if)#exit

R1(config)#interface fastEthernet 0/0.10

R1(config-subif)#encapsulation dot1Q 10

R1(config-subif)#ip address 172.16.10.1 255.255.255.0

R1(config-subif)#exit

R1(config)#interface fastEthernet 0/0.20

R1(config-subif)#encapsulation dot1Q 20

R1(config-subif)#ip address 172.16.20.1 255.255.255.0

R1(config-subif)#end

R1#show ip route

Test the connectivity using:

a. ping <target IP>

b. tracert <target IP>

Step 8: In the simulation mode, using *ping* command, analyze the communication between hosts in the same VLAN and between hosts in different VLANs

CHAPTER 11: LAYER 2 NETWORKS, SPANNING TREE PROTOCOL, LINK AGGREGATION AND ETHERCHANNEL

1. Objectives

At the end of the lab, students will be able to explain the functions of the switches, the Spanning-tree and EtherChannel operation, and to configure Layer 2 networks.

2. Theoretical considerations

The current practical work focuses on the Data Link layer of the ISO/OSI stack (Figure 11.1).

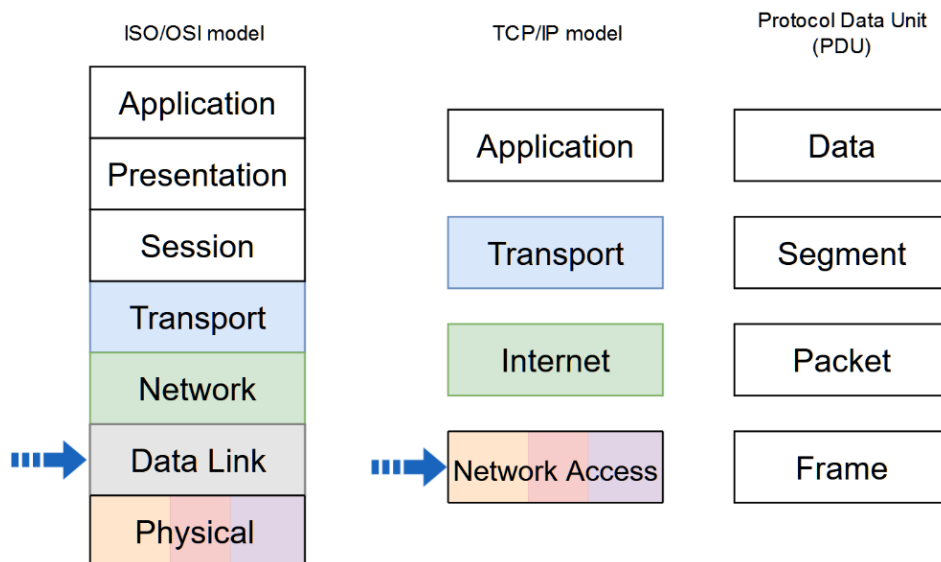


Figure 11.1 Network stack models and PDU naming in each level. The arrows indicate the addressed layers in the current activity

2.1 Switches and bridges

Switches and bridges are layer 2 devices that are used to increase available bandwidth and reduce network congestion. Switches and bridges perform two basic operations: switching data frames and maintaining switching operations. Switches and bridges segment the LAN creating multiple smaller collision domains. Each port creates one segment which is a collision domain because the switch or the bridge learns the MAC addresses of devices connected to each port, enters this information into a switching or bridging table and forwards or blocks traffic based on that table (Figure 11.2). Segmentation allows network congestion to be significantly reduced within each segment. The devices within that segment share the total available bandwidth. If the switch or the bridge does not know where to send the frame, it broadcasts the frame out to all ports. When a reply is returned, the switch or the bridge records the new address in the switching or bridging table. Another advantage of the switched connection is that it permits full-duplex Ethernet which allows the transmission of a packet and the reception of a different

packet at the same time. The disadvantage of layer 2 devices is that they forward broadcast frames to all connected devices on the network, so all hosts connected to the switch or bridge are still part of the same broadcast domain.

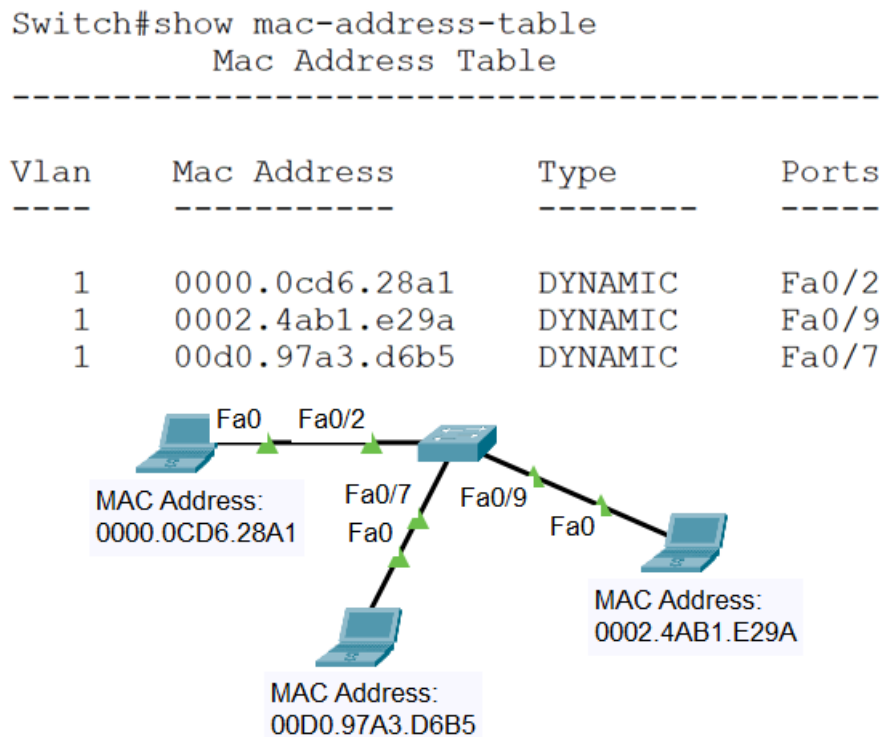


Figure 11.2 Switching table

Switching is classified as symmetric or asymmetric. Symmetric switching provides switched connections between ports with the same bandwidth. Asymmetric switching provides switched connections between ports of unlike bandwidth. Asymmetric switching enables more bandwidth to be dedicated to the server switch port in order to prevent a bottleneck.

Switching modes are classified as store-and-forward or cut-through, each mode representing a compromise between latency and error detection. In store-and-forward switching mode the entire frame is received before any forwarding takes place. This switching mode increases the transmission latency and allows more error detection. In cut-through switching mode the frame is forwarded through the switch before the entire frame is received. At least the frame destination address must be read before the frame can be forwarded. This switching mode decreases the transmission latency and allows less error detection. Cut-through switching mode has two forms: fast-forward and fragment-free. Fast-forward switching forwards the packet after reading the destination address. This switching mode has the lowest level of latency and error detection. Fragment-free switching forwards the packet after reading the first 64 bytes of the frame. Because collision fragments are smaller than 64 bytes, fragment-free switching mode filters out this type of error which also represents the majority of packet errors. This switching mode has a higher level of latency and error detection than the fast-forward mode.

2.2 Spanning-Tree Protocol

Redundant networking topologies increase the reliability of the network by introducing redundant links. These connections introduce physical loops into the network. Because layer 2 has no mechanism to eliminate lost frames, the frames can loop forever in a layer 2 looped

topology causing two types of problems to appear: broadcast storm and switching or bridging table instability. The broadcast storm is created by endlessly flooded broadcast frames to all ports of the switches or bridges, wasting the bandwidth or making the network unusable. Switching or bridging table instability appears when multiple copy of a frame arrive at different ports of a switch or a bridge causing MAC entry instability in the switching or bridging table. The IEEE 802.1d Spanning-Tree Protocol uses the spanning-tree algorithm to create loop free shortest path logical topology in a layer 2 looped topology. The IEEE 802.1w Rapid Spanning-Tree Protocol uses a rapid spanning-tree algorithm to perform the same function as spanning-tree algorithm with a shorter convergence time.

Spanning-Tree Protocol uses Bridge Protocol Data Unit (BPDU) multicast layer 2 messages which are sent by the network devices every two seconds by default. The structure of these messages is presented in Figure 11.3.

Root BID	Root Path Cost	Sender BID	Port ID
----------	----------------	------------	---------

Figure 11.3 BPDU message structure

The BID is an 8-byte field. The two high order bytes are the bridge or switch priority that defaults to 32768 and the six low order bytes are the MAC address of the bridge or switch. The BID structure is presented in Figure 11.4.

7	6	5	4	3	2	1	0
Bridge Priority				MAC address			

Figure 11.4 BID structure

Spanning-Tree Protocol calculates the shortest path network based on cumulative link costs. Link costs are based on the speed of the link. Some of the link costs defined by the IEEE 802.1D standard are presented in Table 11.1.

Table 11.1 Link costs defined by the IEEE 802.1D standard

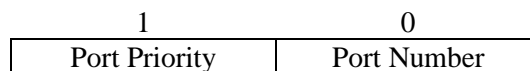
Link Speed	Cost
4Mbps	250
10Mbps	100
16Mbps	62
100Mbps	19
1Gbps	4
10Gbps	2

Some of the link costs defined by the IEEE 802.1w standard are presented in Table 11.2.

Table 11.2 Link costs defined by the IEEE 802.1w standard

Link Speed	Cost
10Mbps	2000000
100Mbps	200000
1Gbps	20000
10Gbps	2000
1Tbps	20
10Tbps	2

The Port ID is a 2-byte field. The high order byte is the port priority that defaults to 128 and the low order byte is the port number. The Port ID structure is presented in Figure 11.5.

**Figure 11.5** Port ID structure

The Spanning-Tree Protocol establishes a single root node, called root bridge and constructs a topology that has one path for reaching every network node. The resulting tree originates from the root bridge. The bridges and switches calculate the shortest path from itself to the root bridge. The first decision that all bridges or switches in the network make is the root bridge identification, which is done through BPDU messages that are received by all bridges and switches. All other decisions in the network are made regarding this root bridge. When a bridge or switch first starts up, it assumes it is the root and sends BPDU-s containing the bridge or switch MAC address in both the root and sender BID. If a bridge or switch receives a BPDU with a lower root BID it sets this root BID in the BPDU-s that are sent out. The bridge or switch with the smallest BID value will be the root bridge. Setting the bridge or switch priority to a smaller value than the default will make the BID smaller and will influence the root bridge identification. For each LAN segment, Spanning-Tree Protocol establishes a designated switch as the closest one to the root bridge which handles all communication from that LAN towards the root bridge. For each non-root bridge a root port is elected, which gives the best path to the root bridge. So, the port with the lowest path cost to the root bridge is elected as the root port. If multiple ports have the same path cost to the root bridge, the port with lowest Port ID is selected as the root port. The Spanning-Tree Protocol also selects the designated ports which are part of the shortest path tree. So, the port with the lowest path cost to the root bridge is selected as the designated port. If more than one port in the segment has the same path cost, the port on which the bridge or the switch has the lowest bridge or switch ID is selected as designated port. On the root bridge, all its ports are designated ports. Redundant links that are not part of the shortest path tree are blocked and data frames received on blocked links are dropped.

Each port on a bridge or switch that is using the Spanning-Tree Protocol has one of the following five states: blocking, listening, learning, forwarding and disabled. In the blocking state ports can only receive BPDU-s, data frames are discarded and no addresses can be learned. It may take up to 20 seconds to change from this state. Ports go from the blocking state to the listening state. In this state, the switches or bridges determine if there are any other paths to the

root bridge. The path that is not the least cost path to the root bridge goes back to the blocked state. In the listening state BPDU-s are processed, user data is not being forwarded and MAC addresses are not being learned. The listening period is called the forward delay and lasts for 15 seconds. Ports go from the listening to the learning state. In this state BPDU-s are processed, user data is not being forwarded, but MAC addresses are learned from any traffic that is seen. The learning state is also called the forward delay and lasts for 15 seconds. A port goes from the learning state to the forwarding state. In this state BPDU-s are processed, user data is forwarded and MAC addresses continue to be learned. The port can be in disabled state when it is administratively down or fails. The time values given for each state are the default values. These values have been calculated on an assumption that there will be a maximum of seven switches in any branch of the spanning tree from the root bridge. When the network topology changes, switches and bridges recompute the Spanning Tree. Convergence on a new spanning-tree topology using the IEEE 802.1D standard can take up to 50 seconds.

2.3 EtherChannel

Link aggregation is the ability to create one logical link using multiple physical links between two devices. EtherChannel is a form of link aggregation used in switched networks (Figure 11.6). This allows for redundancy and higher bandwidth through load sharing among the physical links. EtherChannel creates a one-to-one relationship; that is, one EtherChannel link connects only two devices. An EtherChannel link can be created between two switches or an EtherChannel link can be created between an EtherChannel-enabled server and a switch.

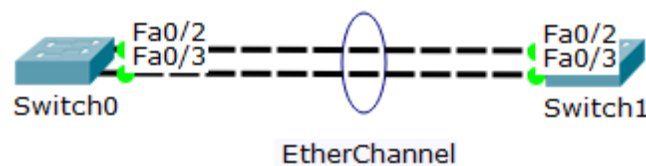


Figure 11.6 *EtherChannel*

EtherChannel technology was originally developed by Cisco as a LAN switch-to-switch technique of grouping several physical ports into one logical channel. When an EtherChannel is configured, the resulting virtual interface is called a port channel. The physical interfaces are bundled together into a port channel interface (Figure 11.7). Most configuration tasks can be done on the EtherChannel interface instead of on each individual port, ensuring configuration consistency throughout the links.

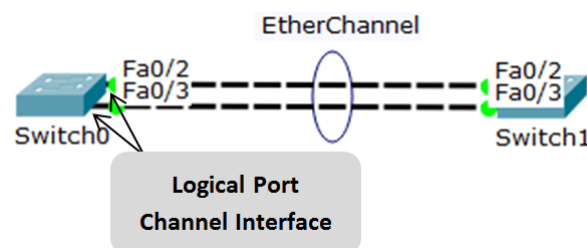


Figure 11.7 *Port channel interface*

EtherChannel relies on existing switch ports. There is no need to upgrade the link to a faster and more expensive connection to have more bandwidth.

Load balancing takes place between links that are part of the same EtherChannel. Depending on the hardware platform, one or more load-balancing methods can be implemented. These methods include source MAC to destination MAC load balancing, or source IP to destination IP load balancing, across the physical links.

EtherChannel creates an aggregation that is seen as one logical link. Where there is only one EtherChannel link, all physical links in the EtherChannel are active because STP sees only one logical link.

EtherChannel provides redundancy because the overall link is seen as one logical connection. Additionally, the loss of one physical link within the channel does not create a change in the topology; therefore a spanning-tree recalculation is not required. Assuming at least one physical link is present, the EtherChannel remains functional, even if its overall throughput decreases because of a lost link within the EtherChannel. The spanning-tree cost is calculated based on the number of ports assigned to the port-channel and it does not dynamically change when links go down or are brought back up within the port-channel. Spanning-Tree Protocol calculates the shortest path network based on cumulative link costs. Link costs are based on the speed of the link. Some of the link costs for links defined by the IEEE 802.1D standard are presented in Table 11.3.

Table 11.3 *Link costs defined by the IEEE 802.1D standard*

Link Speed	Cost (Short mode – 16bit)
10Mbps	100
100Mbps	19
Two-port * 100Mbps EtherChannel	9
Three-port * 100Mbps EtherChannel	8
Four-port * 100Mbps EtherChannel	7
Five-port * 100Mbps EtherChannel	6
Six-port * 100Mbps EtherChannel	5
Seven-port * 100Mbps EtherChannel	5
Eight-port * 100Mbps EtherChannel	5
1Gbps	4
Two-port * 1Gbps EtherChannel	3
Three-port * 1Gbps EtherChannel	2
Four-port * 1Gbps EtherChannel	2
Five-port * 1Gbps EtherChannel	2
Six-port * 1Gbps EtherChannel	2
Seven-port * 1Gbps EtherChannel	2
Eight-port * 1Gbps EtherChannel	1
10Gbps	2
Two-port * 10Gbps EtherChannel	1

Interface types cannot be mixed; they must be compatible-configured Ethernet ports. The individual EtherChannel group member port configuration must be consistent on both devices. If the physical ports of one side are configured as trunks, the physical ports of the other side must also be configured as trunks within the same native VLAN. Additionally, all ports in each EtherChannel link must be configured as Layer 2 ports. Each EtherChannel has a logical port channel interface. A configuration applied to the port channel interface affects all physical interfaces that are assigned to that interface. Layer 3 EtherChannels can be configured on Cisco

Catalyst multilayer switches. A Layer 3 EtherChannel has a single IP address associated with the logical aggregation of switch ports in the EtherChannel.

The maximum number of physical ports in an EtherChannel link depends on the switch hardware platform and IOS version. Usually each EtherChannel can consist of up to 8 compatible-configured Ethernet ports.

The maximum number of EtherChannels supported by a switch depends on the hardware platform and IOS version. The Cisco IOS switch can usually support 6 EtherChannels.

EtherChannel can be configured static, unconditional or it can be formed through negotiation using one of two protocols: Port Aggregation Protocol (PAgP) or Link Aggregation Control Protocol (LACP). These protocols allow ports with similar characteristics to form a channel through dynamic negotiation with adjoining switches.

PAgP is a Cisco-proprietary protocol that aids in the automatic creation and management of EtherChannel links. There are three modes for PAgP: on, desirable and auto. The on mode forces the interface to channel without PAgP. The desirable mode places an interface in an active negotiating state in which the interface initiates negotiations with other interfaces by sending PAgP packets. The auto mode places an interface in a passive negotiating state in which the interface responds to the PAgP packets that it receives, but does not initiate PAgP negotiation. Figure 11.8 presents the channel establishment when ports of switches S1 and S2 are in the different modes for PAgP.

S1	S2	Channel Establishment
On	On	Yes
Auto/Desirable	Desirable	Yes
On/Auto/Desirable	Not Configured	No
On	Desirable	No
Auto/On	Auto	No

Figure 11.8 Channel establishment with PAgP

LACP is part of an IEEE specification (802.3ad) that allows several physical ports to be bundled to form a single logical channel. LACP is also defined in IEEE 802.1AX standard for local and metropolitan area networks. LACP allows a switch to negotiate an automatic bundle by sending LACP packets to the peer. It performs a function similar to PAgP. Because LACP is an IEEE standard, it can be used to facilitate EtherChannels in multi vendor environments. There are three modes for LACP: on, active and passive. The on mode forces the interface to channel without LACP. The active mode places a port in an active negotiating state in which the port initiates negotiations with other ports by sending LACP packets. The passive mode places a port in a passive negotiating state in which the port responds to the LACP packets that it receives, but does not initiate LACP packet negotiation.

Figure 11.9 presents the channel establishment when ports of switches S1 and S2 are in the different modes for LACP.

S1	S2	Channel Establishment
On	On	Yes
Active/Passive	Active	Yes
On/Active/Passive	Not Configured	No
On	Active	No
Passive/On	Passive	No

Figure 11.9 Channel establishment with LACP

3. Lab activity

3.1 Discuss the theoretical aspects presented in this chapter.

3.2 Switch configuration

Cisco switches and routers use a very similar command-line interface (CLI) which is used for configuration and verification purposes.

The help command is question mark (?) which displays the list of commands available for the current command mode, the list of commands that begin with a particular character sequence or the list of keywords or arguments that are associated with a particular command.

Switches have several command modes. The User EXEC mode has a limited command set that can change terminal settings, perform basic tests, or display system information. The *enable* command is used to change from User EXEC mode to Privileged EXEC mode. The Privileged EXEC mode has a larger command set that includes the User EXEC mode command set and the *configure* command used to change from Privileged EXEC mode to global configuration mode. Global configuration mode allows other command modes to be accessed, which are used to configure the switch. The command *exit* is used to exit back from a command mode.

Issue *show running-config* command to view the current configuration file of the switch.

Enter the Privileged EXEC mode with the *enable* command.

Issue *copy running-config startup-config* command to copy the current configuration file to back up configuration file.

In order to completely erase the switch configuration, the following steps have to be followed:

Delete the VLAN database file *vlan.dat* from the flash directory with the *delete flash:vlan.dat* command.

Erase the backup configuration file *startup-config* with the *erase startup-config* command.

Reload the switch with the *reload* command.

```
Switch# enable
Switch# copy running-config startup-config
Switch# delete flash:vlan.dat
Switch# erase startup-config
```



```
Switch# reload
```

Change from Privileged EXEC mode to global configuration mode with the *configure terminal* command.

Set the switch name with the *hostname host_name* command.

```
Switch# enable
Switch# configure terminal
Switch(config)# hostname SWITCH_EXAMPLE
```

Configure the primary terminal line with the following commands:

```
Switch(config)# line console 0
Switch(config-line)# password password
Switch(config-line)# login
Switch(config-line)# exit
```

Configure virtual terminal with the following commands:

```
Switch(config)# line vty 0 4
Switch(config-line)# password secret_password
Switch(config-line)# login
Switch(config-line)# exit
```

•

In order to allow the switch to be accessible by TCP/IP applications, IP addresses and a default gateway should be set. This allows switch configuration using a telnet or ssh connection. Unlike routers, in the case of switches, IP addresses are configured on VLAN interfaces. Configure IP address and default gateway with the following commands:

```
Switch(config)# interface VLAN1
Switch(config-if)# ip address ip_address netmask
Switch(config-if)# no shutdown
Switch(config-if)# exit
Switch(config)# ip default-gateway default_gateway_address
```

3.3 Spanning Tree

Step 1: Configure the network presented in Figure 11.10. If the amber port LED in your topology is not in the same position as in the picture, move the switches so that the amber port LED to be as it is in the figure.

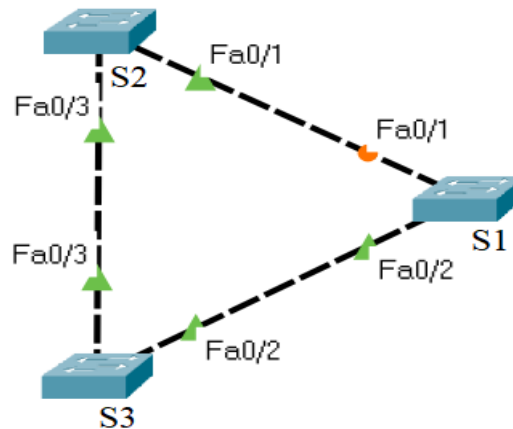


Figure 11.10 Test network topology

1. Specify the host names for the switches
2. Examine the Spanning Tree configuration
 - The port LED color is green if the port is in forwarding state, while the port LED color is amber if the port is in blocking state.
 - View the Spanning Tree information on each switch, with the corresponding show command. Examine and explain the output of the command.

General syntax:

Switch# show spanning-tree

Description: Displays Spanning Tree information

3. Answer the following questions:
 - Why is this topology useful and implemented in computer networks?
 - Which switch is the root bridge and why?

Step 2: To the previous topology connect the users to S2 and S3 switches and add the router that connects the network to other networks like in the Figure 11.11. In this way, an extended star topology with a backup path is obtained.

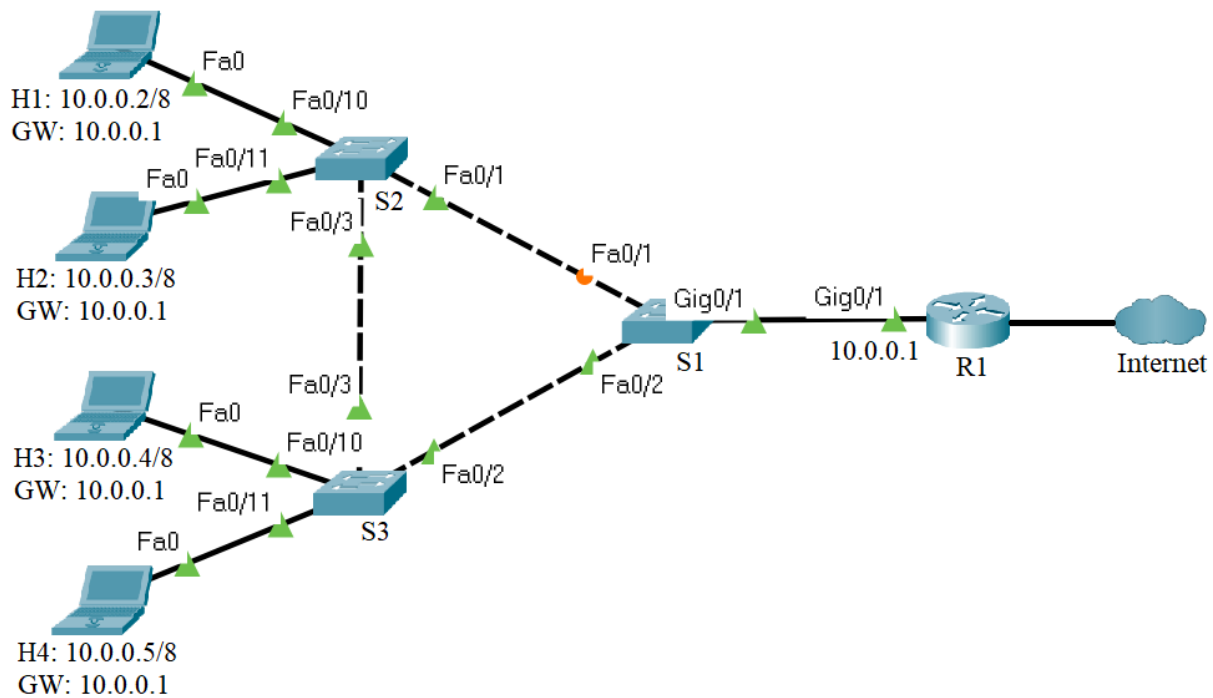


Figure 11.11 Test network topology

Before configuring the network devices, discuss the IPv4 address assignment in the Table 11.4:

Table 11.4 IPv4 addresses for the test network

Device	Interface	IP Address	Netmask	Gateway
Laptop H1	Fa0	10.0.0.2	255.0.0.0	10.0.0.1
Laptop H2	Fa0	10.0.0.3	255.0.0.0	10.0.0.1
Laptop H3	Fa0	10.0.0.4	255.0.0.0	10.0.0.1
Laptop H4	Fa0	10.0.0.5	255.0.0.0	10.0.0.1
R1	Gig0/1	10.0.0.1	255.0.0.0	-

1. Specify the host name for the router
2. Assign the IP information to the hosts and the router
3. Test the connectivity between the hosts and the router using the *ping* command
4. Analyze the network and answer the following questions:
 - Which path should be the backup path (redundant link) and why?
 - Which switch should be the root bridge to obtain the optimal paths in the Layer 2 network?
 - What configuration should be done so that a particular switch becomes the root bridge regardless of the MAC addresses of the switches on the network?
5. Change the root switch by changing its priority to a lower value than the default value

General syntax:

Switch(config)# spanning-tree vlan vlan_number priority priority_number

Description: Changes the Spanning-tree priority of the switch

Consider: switch *S1*, vlan *1* and priority *0*

6. Issue *show spanning-tree* command (Figure 11.12) several times to view the Spanning Tree information on switch *S1*

- Pay attention to the following:
 - Switch *S1* becomes the root bridge
 - The port in the blocking state goes to the forwarding state passing through listening and learning states

```

S1#show spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    1
            Address    00E0.F7B4.DDC2
            This bridge is the root
            Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    1 (priority 0 sys-id-ext 1)
            Address    00E0.F7B4.DDC2
            Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec
            Aging Time 20

Interface          Role Sts Cost          Prio.Nbr Type
-----
Fa0/1              Desg LRN 19           128.1   P2p
Fa0/2              Desg FWD 19           128.2   P2p
Gi0/1              Desg FWD 4            128.25  P2p

```

Figure 11.12 *Show spanning-tree command output for S1*

- In the network topology, the amber port LED becomes green while a port LED between *S2* and *S3* switches becomes amber (Figure 11.13). The link between *S1* and *S2* forwards the traffic while the link between *S2* and *S3* becomes the backup path, the redundant link. Issue *show spanning-tree* command (Figure 11.14) to view the Spanning Tree information on the switch having the amber port LED.

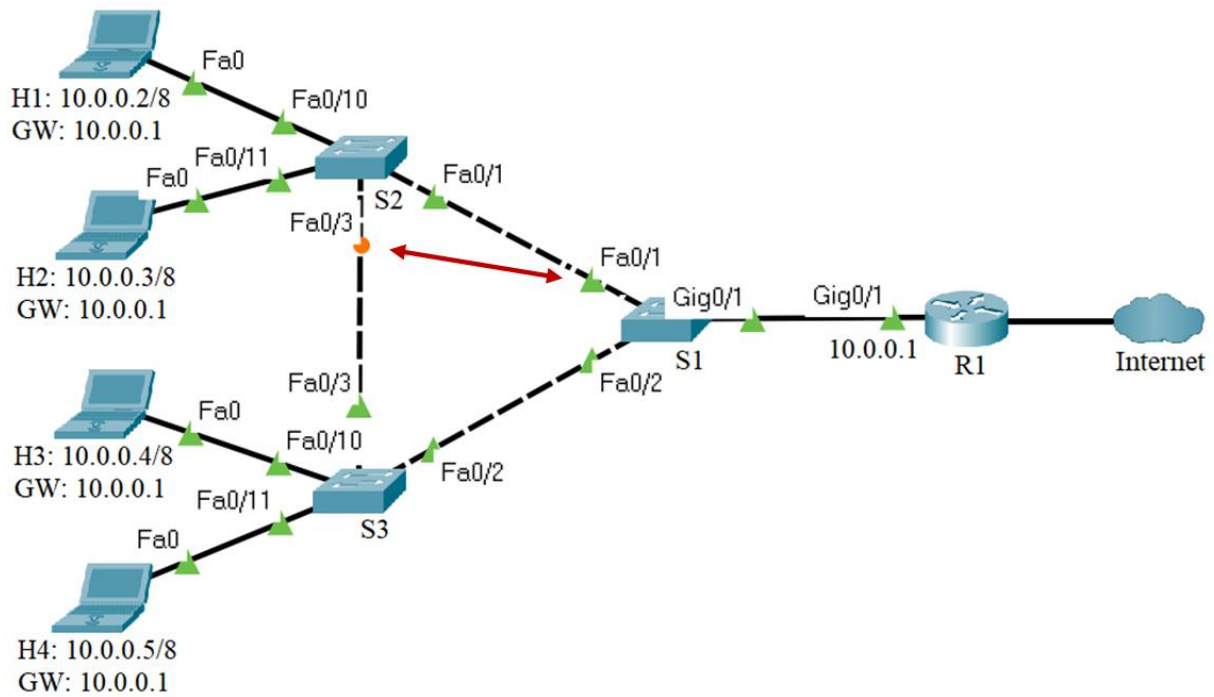


Figure 11.13 STP changes the state of the ports

```
S2#show spanning-tree
VLAN0001
Spanning tree enabled protocol ieee
Root ID    Priority    1
           Address    00E0.F7B4.DDC2
           Cost      19
           Port      1(FastEthernet0/1)
           Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec

Bridge ID  Priority    32769 (priority 32768 sys-id-ext 1)
           Address    00D0.D349.4CEC
           Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec
           Aging Time 20

Interface      Role Sts Cost      Prio.Nbr Type
-----
Fa0/1          Root FWD 19         128.1    P2p
Fa0/3          Altn BLK 19         128.3    P2p
Fa0/10         Desg FWD 19         128.10   P2p
Fa0/11         Desg FWD 19         128.11   P2p
```

Figure 11.14 Show spanning-tree command output for the switch having the amber port LED

7. Continuously test the connectivity between the host H1 and the router using the *ping* command with *-t* option (Figure 11.15)

```

C:\>ping
Packet Tracer PC Ping

Usage: ping [-n count | -v TOS | -t ] target

C:\>ping -t 10.0.0.1

Pinging 10.0.0.1 with 32 bytes of data:

Reply from 10.0.0.1: bytes=32 time<1ms TTL=255
Reply from 10.0.0.1: bytes=32 time<1ms TTL=255
Reply from 10.0.0.1: bytes=32 time<1ms TTL=255
Reply from 10.0.0.1: bytes=32 time<1ms TTL=255
Reply from 10.0.0.1: bytes=32 time<1ms TTL=255
Reply from 10.0.0.1: bytes=32 time<1ms TTL=255
Reply from 10.0.0.1: bytes=32 time<1ms TTL=255
Reply from 10.0.0.1: bytes=32 time<1ms TTL=255
Reply from 10.0.0.1: bytes=32 time<1ms TTL=255
Reply from 10.0.0.1: bytes=32 time<1ms TTL=255

```

Figure 11.15 Connectivity test between the host H1 and the router

8. While the *ping* command is testing the connectivity between the host H1 and the router, remove the link between the S1 and the S2 switches (Figure 11.16).

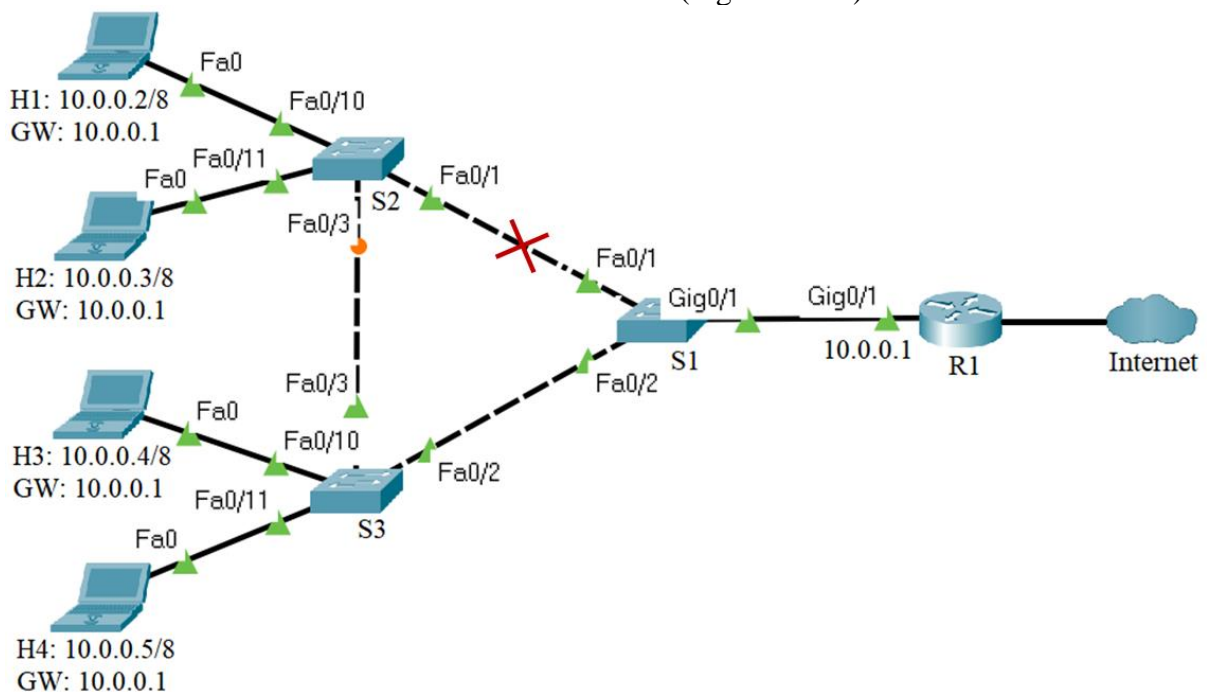


Figure 11.16 Removing the link between S1 and S2 switches

- Issue *show spanning-tree* command (Figure 11.17) several times to view the Spanning Tree information on switch S2; pay attention to the port in the blocking state, it goes to the forwarding state passing through listening and learning states

```
S2#show spanning-tree
VLAN0001
Spanning tree enabled protocol ieee
Root ID      Priority      1
             Address      00E0.F7B4.DDC2
             Cost        38
             Port        3(FastEthernet0/3)
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec

Bridge ID    Priority      32769 (priority 32768 sys-id-ext 1)
             Address      00D0.D349.4CEC
             Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time  20

Interface      Role Sts Cost          Prio.Nbr Type
-----
Fa0/3          Root LSN 19           128.3    P2p
Fa0/10         Desg FWD 19           128.10   P2p
Fa0/11         Desg FWD 19           128.11   P2p
```

Figure 11.17 Show spanning-tree command output for S2

- In the network topology, the amber port LED becomes green (Figure 11.18), the link between S2 and S3 forwards the traffic.

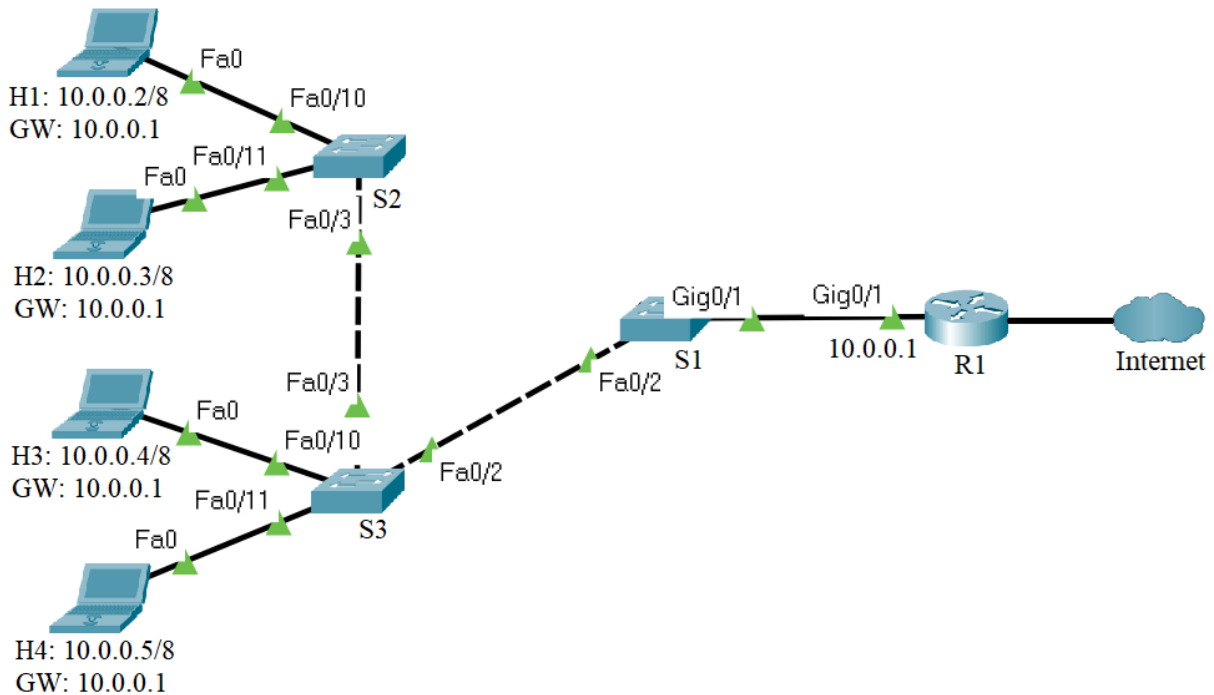


Figure 11.18 STP changes the state of the ports

- The Spanning-tree protocol restores connectivity between the H1 host and the router through the redundant link (Figure 11.19)

```

Reply from 10.0.0.1: bytes=32 time<1ms TTL=255
Reply from 10.0.0.1: bytes=32 time<1ms TTL=255
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Reply from 10.0.0.1: bytes=32 time=27ms TTL=255
Reply from 10.0.0.1: bytes=32 time<1ms TTL=255
Reply from 10.0.0.1: bytes=32 time<1ms TTL=255
    
```

Figure 11.19 Connectivity test between the host H1 and the router

9. Restore the link between the S1 and the S2 switches and observe how Spanning-tree protocol chooses the shortest paths to the root bridge and blocks the redundant links (Figure 11.20).

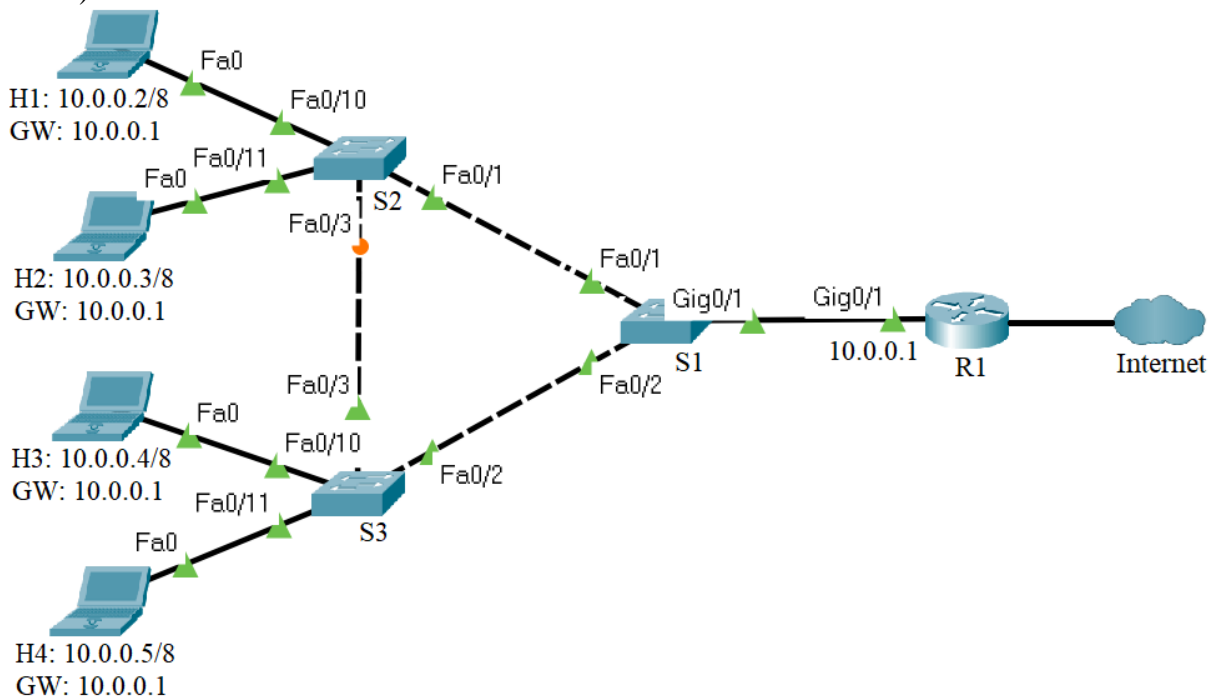


Figure 11.20 Restoring the link between S1 and S2 switches

3.4 EtherChannel

Cable the network presented in the Figure 11.21.

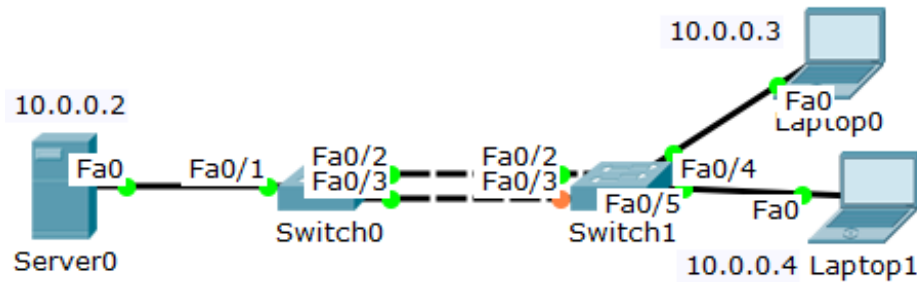


Figure 11.21 Test network topology

Before configuring the network devices, discuss the IPv4 address assignment in the Table 11.5:

Table 11.5 IPv4 addresses for the test network

Device	Interface	IP Address	Netmask
Server0	Fa0	10.0.0.2	255.0.0.0
Laptop 0	Fa0	10.0.0.3	255.0.0.0
Laptop 1	Fa0	10.0.0.4	255.0.0.0

1. Configure the IP addresses on the hosts.
2. Verify the connectivity between the laptops and Server0 with the `ping` command.
3. Connect to the Switch0 and enter the Privileged EXEC mode. View the Spanning Tree information with the `show spanning-tree` command. Examine and explain the output of this command.

```
Switch0#show spanning-tree
```

4. Repeat the previous step for Switch1.

5. Connect to Switch0 and specify the interfaces that compose the EtherChannel group using the `interface range` interface global configuration mode command. Create the port channel interface with the `channel-group identifier mode on` command in interface range configuration mode. The identifier specifies a channel group number.

```
Switch0(config)#interface range fastEthernet 0/2-3
```

```
Switch0(config-if-range)#channel-group 1 mode on
```

6. Repeat the previous step for Switch1.

7. Connect to the Switch0 and enter the Privileged EXEC mode. View the running-config file with the `show running-config` command. Examine and explain the output of this command. View the Etherchannel information with the `show etherchannel summary` command. Examine and explain the output of this command. View the Spanning Tree information with the `show spanning-tree` command. Examine and explain the output of this command.

```
Switch0#show running-config
```

```
Switch0#show etherchannel summary
```

```
Switch0#show spanning-tree
```

8. Repeat the previous step for Switch1.

9. Connect to Switch0 and enter port channel interface configuration mode using the *interface port-channel* command, followed by the interface identifier. Configure the EtherChannel as a trunk interface using the *switchport mode trunk* command.

```
Switch0(config)#interface port-channel 1
```

```
Switch0(config-if)#switchport mode trunk
```

10. Repeat the previous step for Switch1.

11. Connect to the Switch0 and enter the Privileged EXEC mode. View the running-config file with the *show running-config* command. Examine and explain the output of this command. View the trunking information with the *show interfaces trunk* command. Examine and explain the output of this command.

```
Switch0#show running-config
```

```
Switch0#show interfaces trunk
```

12. Repeat the previous step for Switch1.

13. Connect to Switch0 and configure EtherChannel load balancing method using the *port-channel load-balance* global configuration mode command. Select the load-distribution method based on the destination-host MAC address of the incoming packet (*dst-mac*). Enter the Privileged EXEC mode and view the EtherChannel load balancing method information with the *show etherchannel load-balance* command. Examine and explain the output of this command.

```
Switch0(config)#port-channel load-balance dst-mac
```

```
Switch0(config)#end
```

```
Switch0#show etherchannel load-balance
```

CHAPTER 12: SECURITY THREATS IN COMPUTER NETWORKS

1. Objectives

At the end of the activity, students will be able to understand and analyze common security threats that occur in computer networks.

2. Theoretical considerations

The current practical work focuses on the Data Link, Network and Application layers of the ISO/OSI stack (Figure 12.1).

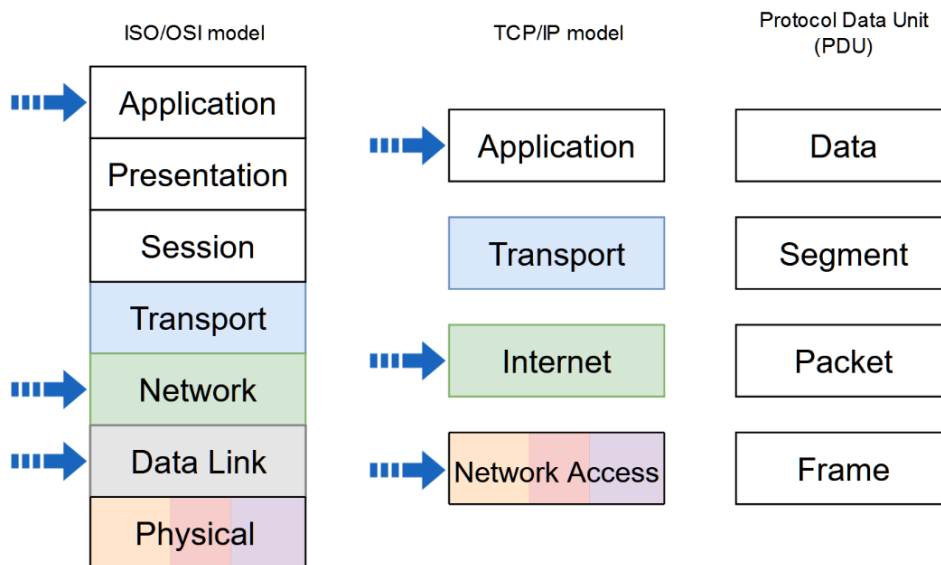


Figure 12.1 Network stack models and PDU naming in each level. The arrows indicate the addressed layers in the current activity

2.1 Common security threats

Network security in computer networking is a very broad domain and the security attacks can have different purposes, such as: service interruption, gaining elevated privilege for various services, data stealing, data corruption, etc. Security threats occur at every layer of the ISO/OSI model and networks must be secured with proper defenses against any possible attack.

The current activity demonstrates the working principles of a few security threats using the Cisco Packet Tracer tool. In a real-life scenario, additional tools might be required to perform these attacks, but the objective of this activity is academic only. The desired purpose is to understand how certain attacks are implemented and what are the best ways to prevent them from taking place.

The main concepts that are addressed in this activity are the following:

- **ARP spoofing:** This is the process in which a malicious device is spoofing its own MAC address, meaning it is masking its own MAC Address with a different MAC address that can belong to a different network device. In order to inform the other devices of the fake MAC address, the malicious device is sending a gratuitous ARP to the other network hosts informing them of the MAC address that resides at a specific IP Address. After each network host receives the ARP request they will store the new pair of IP - MAC addresses in their own ARP cache table and when they will send a packet to the particular device, they will fill the Layer 2 header with the spoofed MAC address
- **Network sniffer:** A network sniffer is a device that can intercept network traffic and records it using traffic monitoring tools
- **Denial of Service (DoS):** This is a type of attack that has the purpose to restrict access to normal network functions
- **Rogue server:** A rogue server does not belong to the institution (or stakeholder) that owns the network. Such a server can offer various services and invalid information to network devices with malicious intent
 - **Rogue web server:** can offer web pages that look like a real website, but they are in fact copies of a real site
 - **Rogue DHCP server:** can offer invalid addressing, e.g. wrong default gateway for denying other hosts access to the internet, wrong DNS server to make hosts access invalid web server
 - **Rogue DNS server:** can provide fake mappings between URL - IP address with the purpose to force users to access a fake web server which apparently resides at a valid URL
- **Phishing:** A type of attack meant to steal information through a fraudulent message or web site

3. Lab activity

3.1 ARP spoofing for DoS and data sniffing.

An example of ARP spoofing attack with the end goal to deny access to certain resources and to allow data sniffing can be seen in the Figure 12.2:

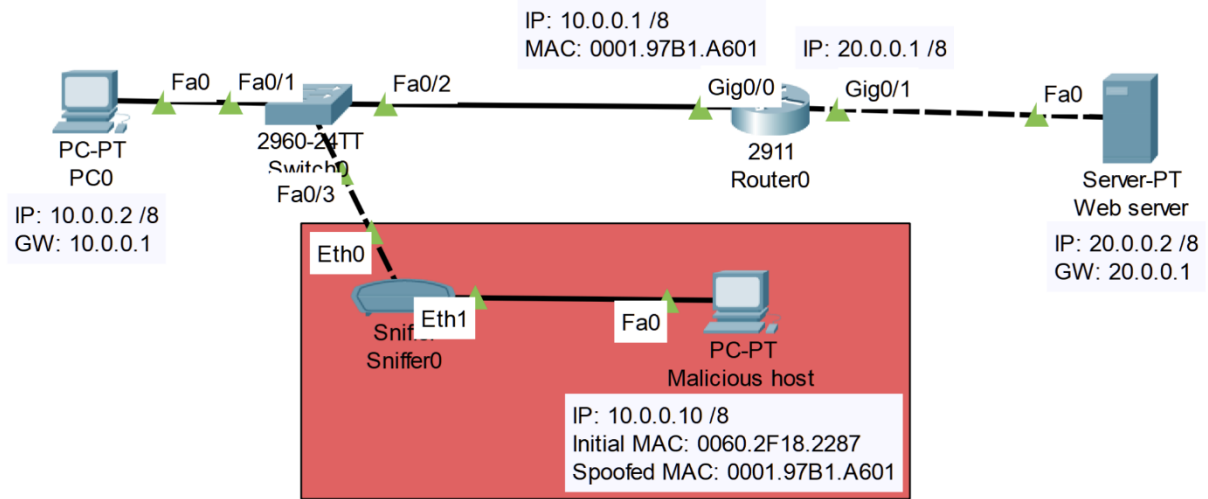
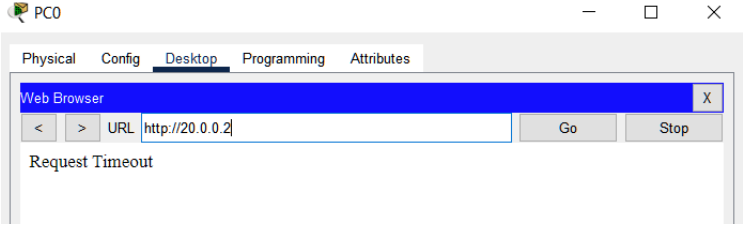
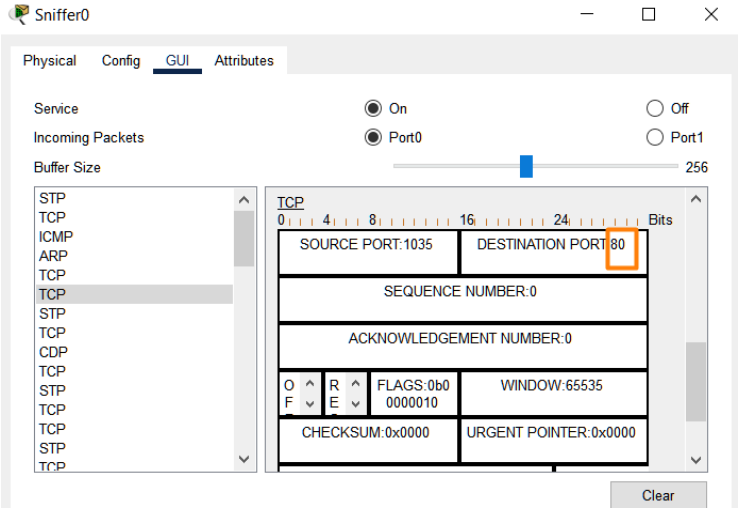


Figure 12.2 Network attack topology: *ARP spoofing for DoS and data sniffing*

In order to perform the attack, configure the topology in Packet Tracer, then follow the steps described below (Table 12.1).

Table 12.1 *ARP spoofing for DoS and data sniffing attack steps*

<p>1. A web server resides at IP address 20.0.0.2. Accessing the default web page from PC0 having the IP address 10.0.0.2, the Cisco Packet Tracer view will look like the image on the right</p>	
<p>2. The figure shows that the router's Gig 0/0 interface has the MAC address: 0001.97B1.A601</p> <p>The malicious host can override its own MAC address with that of the router</p>	
<p>3. The next step for the malicious host is to inform the other network devices in the network that the MAC address of the router actually corresponds to the IP address of the malicious host. This can be done by generating continuous traffic in the network, e.g. using the ping command</p>	<pre>C:\>ping 10.0.0.2 Pinging 10.0.0.2 with 32 bytes of data: Reply from 10.0.0.2: bytes=32 time=2ms TTL=128 Reply from 10.0.0.2: bytes=32 time=5ms TTL=128 Reply from 10.0.0.2: bytes=32 time=4ms TTL=128</pre>

	<p>with the -t parameter</p>
<p>4. Next, the ARP cache entries on the other network devices can be verified (on PC0 having the IP address 10.0.0.2 and on the switch)</p> <p>Viewing this information it can be seen that the computer will add the same MAC address when generating traffic towards the gateway of the malicious host, but the switch will as well redirect the traffic on the Fa 0/3 interface which links towards the malicious host (if the MAC address table does not change, use the <i>#clear mac-address-table</i> command)</p>	<pre>C:\>arp -a Internet Address Physical Address Type 10.0.0.1 0001.97b1.a601 dynamic 10.0.0.10 0001.97b1.a601 dynamic</pre> <pre>Switch#show mac-address-table Mac Address Table ----- Vlan Mac Address Type Ports ---- - 1 0001.97b1.a601 DYNAMIC Fa0/3 1 0001.c98c.5a65 DYNAMIC Fa0/1</pre>
<p>5. Next, when PC0, having IP address 10.0.0.2, tries to access the web server, it will create packet having the correct MAC address of the network gateway (interface Gig 0/0 on the router), but the switch will redirect this packet towards the malicious host through the network sniffer</p>	
<p>6. The sniffer can also be opened and inspect its GUI. The TCP traffic that is generated from the computer towards the web server can be inspected. Not much information is seen in this Cisco Packet Tracer example, but a real life test can reveal multiple traffic flows being generated from the targeted PC</p>	

After analyzing the entire sequence of steps, the ARP spoofing attack was successful with the outcome of denying the service to the web server and eavesdropping on the traffic generated by the computer.

Possible ways to overcome these security threats include (but are not limited to):

- Limiting the number of allowed MAC addresses per switch port
 - Configuring inspection of MAC - IP address consistency
- Research other mechanisms to prevent ARP spoofing.

3.2 ARP spoofing for phishing

An example of a phishing attack from a web server can be seen in the Figure 12.3 below where an attacker is connecting to the network with a router (with a static IP address) and a web phishing server in the network behind the connected router:

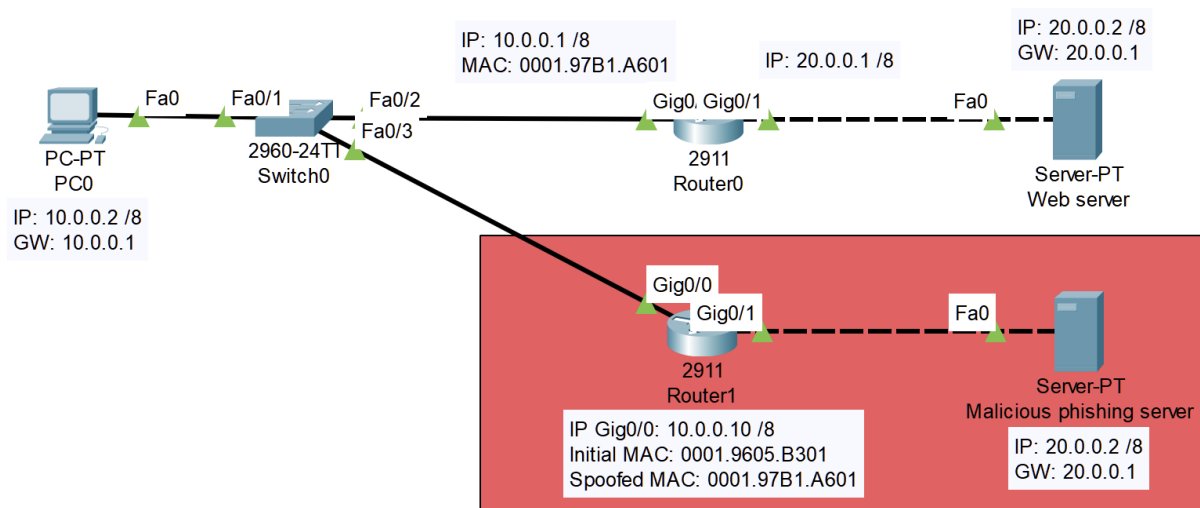
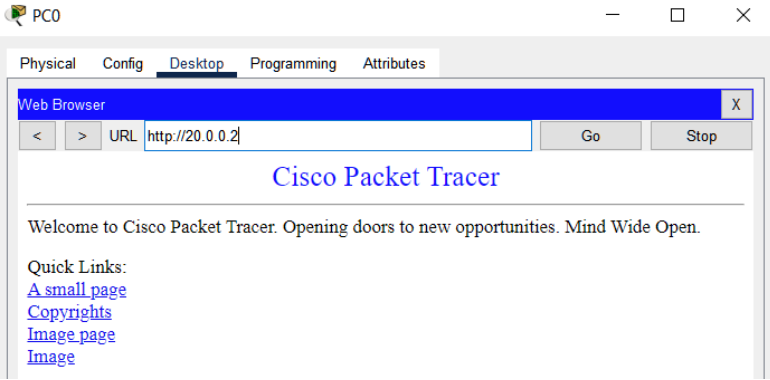
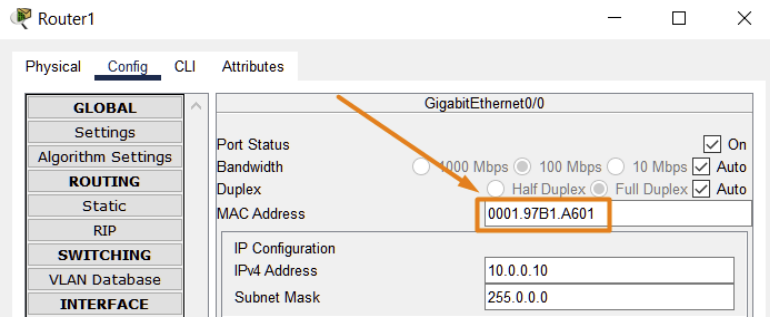
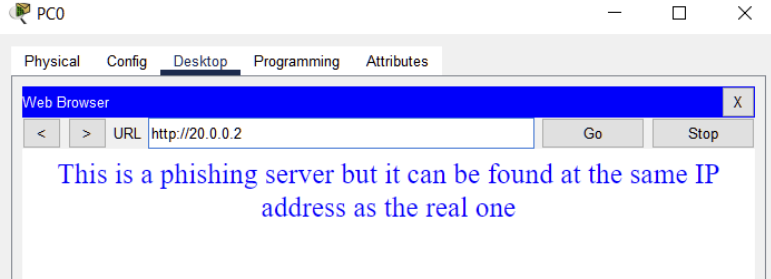


Figure 12.3 Network attack topology: *ARP spoofing for phishing*

In order to perform the attack, configure the topology in Packet Tracer, then follow the steps described below (Table 12.2).

Table 12.2 ARP spoofing for phishing attack steps

<p>1. A web server resides at IP address 20.0.0.2. Accessing the default web page from PC0 having the IP address 10.0.0.2, the Cisco Packet Tracer view will look like the image on the right</p>	
<p>2. The figure shows that Router0's Gig0/0 interface has the MAC address: 0001.97B1.A601</p> <p>The threat actor connects to the network with a router where the MAC address is overridden with the MAC address of Router0</p>	
<p>3. The next step for the threat actor is to inform the other network devices in the network that the MAC address of Router0 actually corresponds to Router1 (the malicious router). This can be done by generating continuous traffic in the network, e.g. using the ping command from the Router1's CLI as shown in the image on the right</p>	<pre>Router1#ping 10.255.255.255 Type escape sequence to abort. Sending 5, 100-byte ICMP Echos to 10.255.255.255, timeout is 2 seconds: Reply to request 0 from 10.0.0.2, 0 ms Reply to request 1 from 10.0.0.2, 0 ms Reply to request 2 from 10.0.0.2, 0 ms Reply to request 3 from 10.0.0.2, 0 ms Reply to request 4 from 10.0.0.2, 0 ms Router1#ping 10.0.0.2 Type escape sequence to abort. Sending 5, 100-byte ICMP Echos to 10.0.0.2, timeout is 2 seconds: !!!! Success rate is 100 percent (5/5), round-trip min/ avg/max = 0/2/13 ms Router1#</pre>

<p>4. Next, the ARP cache entries on the other network devices can be verified (on PC0 having the IP address 10.0.0.2 and on the switch)</p> <p>Viewing this information it can be seen that the computer will add the same MAC address when generating traffic towards the gateway of Router1, but the switch will as well redirect the traffic on the Fa 0/3 interface which links towards Router1 (if the MAC address table does not change, use the <code>#clear mac-address-table</code> command)</p>	<pre>C:\>arp -a Internet Address Physical Address Type 10.0.0.1 0001.97b1.a601 dynamic 10.0.0.10 0001.97b1.a601 dynamic</pre> <pre>Switch#show mac-address-table Mac Address Table ----- Vlan Mac Address Type Ports ----- 1 0001.97b1.a601 DYNAMIC Fa0/3 1 0001.c98c.5a65 DYNAMIC Fa0/1</pre>
<p>5. Next, when PC0 will try to access the web server, it will create a packet having the correct MAC address of the network gateway (interface Gig 0/0 on Router0), but the switch will redirect this packet towards Router1.</p> <p>After the packet reaches Router1, the threat actor has already set up in place a simulated network which mimiques the same IP addresses as in the real network but creates a phishing website showing different content, but which is still accessible on the same IP address</p>	

After analyzing the entire sequence of steps, the ARP spoofing attack was successful with the outcome of making the targeted host access a fake server that can accomplish phishing scenarios if configured to e.g. accept credentials input.

One of the best advised ways to overcome this issue is to prevent/avoid it entirely by not accessing unsecured websites from untrusted networks or by not introducing sensitive credential information when present in an untrusted network.

- Research other mechanisms to prevent ARP spoofing and phishing.

3.3 Rogue DHCP and DNS servers for phishing

An example of a phishing attack which is performed with the help of a rogue DHCP and DNS server can be seen in the Figure 12.4:

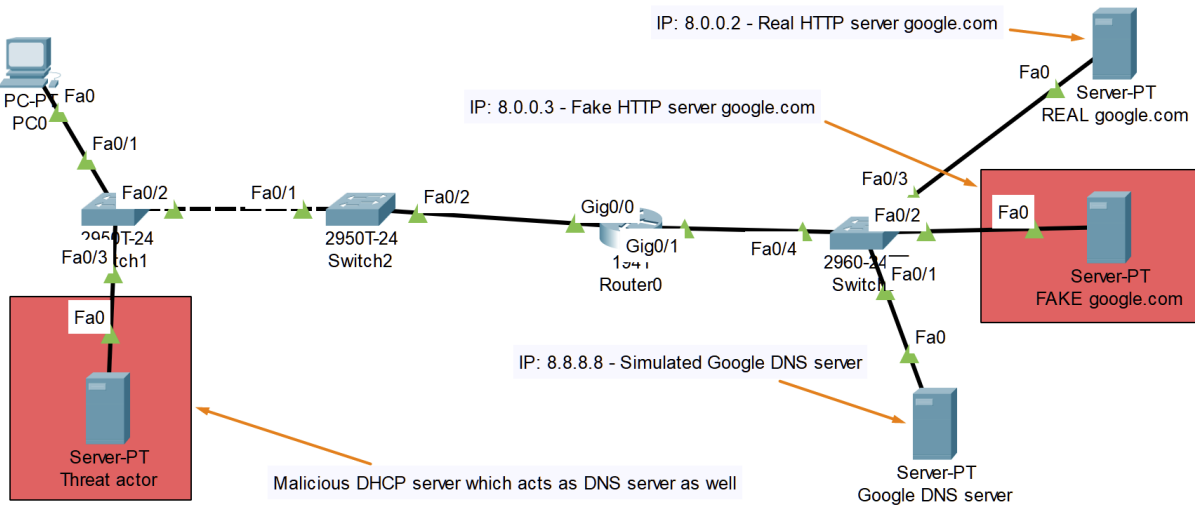
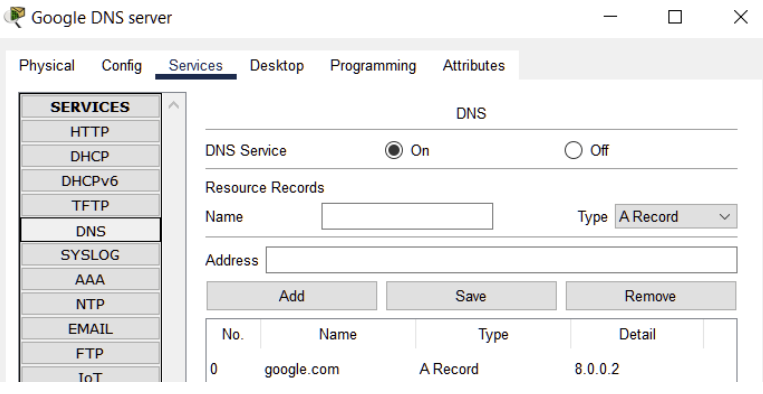

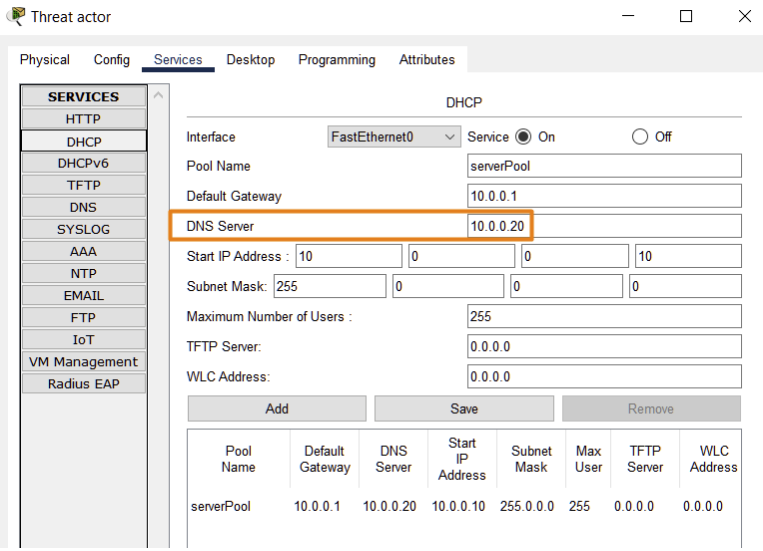


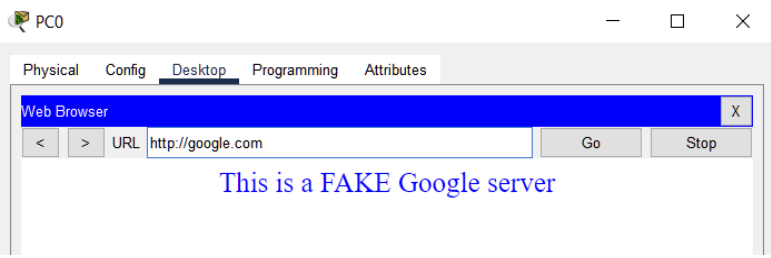
Figure 12.4 Network attack topology: *Rogue DHCP and DNS servers for phishing*

The threat actor connects to the network with a server providing false DHCP addressing information (the DNS server configuration being the most important addressing information in this example). When the computers inside the network use the false DNS server to access the IP address of a web server, they will be redirected to the phishing server instead of the real one.

Configure the topology in Packet Tracer, then follow the steps described below (Table 12.3).

Table 12.3 *Rogue DHCP and DNS servers for phishing attack steps*

<p>1. Initial configuration steps:</p> <ul style="list-style-type: none"> • Leave the threat actor unconfigured (or remove its link to the switch) • Configure DHCP on Router0 and ensure that it provides a valid DNS server in the addressing information (as seen on the right) • Configure the simulated Google DNS server as seen on the right side 	<pre>ip dhcp pool pool1 network 10.0.0.0 255.0.0.0 default-router 10.0.0.1 dns-server 8.8.8.8</pre> 																
<p>2. A simulated Google server resides at the server having the 8.0.0.2 IP address. Accessing this server from PC0, the Cisco Packet Tracer view will look like the image on the right</p>																	
<p>3. Connect the threat actor to the network and configure its DHCP service as seen in the image on the right side. Notice the different DNS server which in fact corresponds to the threat actor's static IP address.</p>	 <table border="1" data-bbox="925 1646 1492 1747"> <thead> <tr> <th>Pool Name</th> <th>Default Gateway</th> <th>DNS Server</th> <th>Start IP Address</th> <th>Subnet Mask</th> <th>Max User</th> <th>TFTP Server</th> <th>WLC Address</th> </tr> </thead> <tbody> <tr> <td>serverPool</td> <td>10.0.0.1</td> <td>10.0.0.20</td> <td>10.0.0.10</td> <td>255.0.0.0</td> <td>255</td> <td>0.0.0.0</td> <td>0.0.0.0</td> </tr> </tbody> </table>	Pool Name	Default Gateway	DNS Server	Start IP Address	Subnet Mask	Max User	TFTP Server	WLC Address	serverPool	10.0.0.1	10.0.0.20	10.0.0.10	255.0.0.0	255	0.0.0.0	0.0.0.0
Pool Name	Default Gateway	DNS Server	Start IP Address	Subnet Mask	Max User	TFTP Server	WLC Address										
serverPool	10.0.0.1	10.0.0.20	10.0.0.10	255.0.0.0	255	0.0.0.0	0.0.0.0										

<p>4. At some point, PC0 will have to update its addressing information by requesting a new lease from the DHCP server. This step can be manually simulated as seen on the right side.</p> <p>Notice how the DNS server has changed, meaning that PC0 receives the lease from the threat actor and not from Router0.</p> <p>Investigate how this happens by using the Simulation tool provided by Cisco Packet Tracer.</p> <p>It is obvious that the DNS server configured on the threat actor's server will not match the "google.com" URL to the correct IP address, it will match the URL to the fake server's IP address as seen in the topology. Running an <i>nslookup</i> command on PC0 will prove this.</p>	<pre> IP Address.....: 10.0.0.7 Subnet Mask.....: 255.0.0.0 Default Gateway.....: 10.0.0.1 DNS Server.....: 8.8.8.8 C:\>nslookup google.com Server: [8.8.8.8] Address: 8.8.8.8 Non-authoritative answer: Name: google.com Address: 8.0.0.2 Real server C:\>ipconfig /release IP Address.....: 0.0.0.0 Subnet Mask.....: 0.0.0.0 Default Gateway.....: 0.0.0.0 DNS Server.....: 0.0.0.0 C:\>ipconfig /renew IP Address.....: 10.0.0.14 Subnet Mask.....: 255.0.0.0 Default Gateway.....: 10.0.0.1 DNS Server.....: 10.0.0.20 C:\>nslookup google.com Server: [10.0.0.20] Address: 10.0.0.20 Non-authoritative answer: Name: google.com Address: 8.0.0.3 Fake server </pre>
<p>5. After PC0 has been compromised, accessing the google.com web page will redirect to the fake server showing a different page</p>	 <p>The screenshot shows a PC0 desktop environment with a window titled 'Web Browser'. The address bar contains 'http://google.com' and the page content displays 'This is a FAKE Google server' in blue text.</p>

After analyzing the entire sequence of steps, this phishing attack was successful, and it can trick the user into entering his credentials on a fake web page having a seemingly valid URL.

Research mechanisms to prevent rogue servers to provide false network services and mechanisms to prevent phishing attacks.